NLP4DH 2021

Workshop on Natural Language Processing for Digital Humanities

Proceedings of the Workshop

December 19, 2021

Order copies of this and other ACL proceedings from:

Association for Computational Linguistics (ACL)
209 N. Eighth Street
Stroudsburg, PA 18360
USA
Tel: +1-570-476-8006
Fax: +1-570-476-0860
acl@aclweb.org

ISBN 978-952-94-5833-2
Rootroo Oy

Preface

Textual sources are essential for research in digital humanities. Especially when larger datasets are analyzed, the use of natural language processing (NLP) technologies is essential. However, NLP is still often focused to written standard languages, which customarily differs from specific genres and text types that may interest a digital humanist today. The situation is even more complicated when the research is done on minority languages, or historical and dialectal materials.

Natural language processing has usually a strong computer science focus, which means that methods are developed to cater for higher numerical results and to solve some rather abstract level tasks such as machine translation, poem generation or sentiment analysis. Digital humanities, on the other hand, has usually a strong humanities focus which means that the research questions are typically more concrete, diving deeper to understanding some phenomena rather than solving a problem. Natural language processing also seeks to validate the methods, whereas digital humanities takes the validity of the methods for granted. This is due to the fact that a method is often the end goal in natural language processing, where as a method is just a tool in the digital humanities. The two fields work from very different starting points, and therefore we believe that more venues are needed where scholars from both fields can come together and learn from each other.

We believe that digital humanists recognize the shortcomings of the contemporary natural language processing tools, and the NLP community has already come up with various fully functional solutions. However, these communities would benefit from further communication. For example, model fine tuning and retraining are among useful technologies in NLP that could be applied to efficiently improve the result on these divergent varieties. Similarly work in digital humanities often results in open datasets that could be used to compare different strategies. In this workshop we aimed to foster and initiate wider conversation and sharing of examples of how NLP tools are best leveraged to the research questions that are relevant in humanities.

The Workshop on Natural Language Processing for Digital Humanities (NLP4DH) was organized for the first time in December 19, 2021 with ICON 2021: The 18th International Conference on Natural Language Processing. Our workshop received 42 submissions, out of which 21 were accepted to be presented in the workshop. We are especially excited about the upcoming special issue in the Journal of Data Mining & Digital Humanities that will feature extended versions of some of the papers accepted in the workshop.

https://rootroo.com

Organizing Committee

- Mika Hämäläinen, University of Helsinki and Rootroo Ltd
- Khalid Alnajjar, University of Helsinki and Rootroo Ltd
- Niko Partanen, University of Helsinki
- Jack Rueter, University of Helsinki

Program Committee

- Iana Atanassova, Université de Bourgogne Franche-Comté
- Yuri Bizzoni, Aarhus University
- Miriam Butt, University of Konstanz
- Jeremy Bradley, University of Vienna
- Won Ik Cho, Seoul National University
- Stefania Degaetano-Ortlieb, Saarland University
- Quan Duong, University of Helsinki
- Valts Ernštreits, University of Latvia, Livonian Institute
- Luke Gessler, Georgetown University
- Hugo Gonçalo Oliveira, University of Coimbra
- Kenichi Iwatsuki, ARIKTTA
- Maciej Janicki, University of Helsinki
- Heiki-Jaan Kaalep, University of Tartu
- Maximilian Koppatz, Sanoma Media Finland
- Mikko Kurimo, Aalto University
- Leo Leppänen, University of Helsinki
- Enrique Manjavacas Arevalo, University of Leiden
- Matej Martinc, Jozef Stefan Institute
- Flammie Pirinen, UiT The Arctic University of Norway
- Lidia Pivovarova, University of Helsinki
- Tyler Shoemaker, University of California, Davis
- Liisa Lotta Tarvainen-Li, Acolad
- Jörg Tiedemann, University of Helsinki
- Jouni Tuominen, Aalto University

- Linda Wiechetek, UiT The Arctic University of Norway
- Joshua Wilbur, University of Tartu
- Shuo Zhang, Bose Corporation
- Emily Öhman, Waseda University

Table of Contents

Conference Program

Sunday, December 19, 2021

9:45–10:00 ***Opening***

10:00–11:00 **Session 1: Sentiment**

10:00–10:20 *Sentiment Dynamics of Success: Fractal Scaling of Story Arcs Predicts Reader Preferences*
Yuri Bizzoni, Telma Peura, Mads Rosendahl Thomsen and Kristoffer Nielbo

10:20–10:40 *The Validity of Lexicon-based Sentiment Analysis in Interdisciplinary Research*
Emily Öhman

10:40–11:00 *How Does the Hate Speech Corpus Concern Sociolinguistic Discussions? A Case Study on Korean Online News Comments*
Won Ik Cho and Jihyung Moon

11:00–11:15 **Coffee break**

11:15–12:15 **Session 2: Historical data**

11:15–11:35 *MacBERTh: Development and Evaluation of a Historically Pre-trained Language Model for English (1450-1950)*
Enrique Manjavacas Arevalo and Lauren Fonteyn

11:35–11:55 *Named Entity Recognition for French medieval charters*
Sergio Torres Aguilar and Dominique Stutzmann

11:55–12:15 *Processing M.A. Castrén's Materials: Multilingual Historical Typed and Handwritten Manuscripts*
Niko Partanen, Jack Rueter, Khalid Alnajjar and Mika Hämäläinen

Sunday, December 19, 2021 (continued)

12:15–13:15 Lunch

13:15–14:15 Session 3: Literature

13:15–13:35 *Lotte and Annette: A Framework for Finding and Exploring Key Passages in Literary Works*
Frederik Arnold and Robert Jäschke

13:35–13:55 *Using Referring Expression Generation to Model Literary Style*
Nick Montfort, Ardalan SadeghiKivi, Joanne Yuan and Alan Y. Zhu

13:55–14:15 *The concept of nation in nineteenth-century Greek fiction through computational literary analysis*
Fotini Koidaki, Despina Christou, Katerina Tiktopoulou and Grigorios Tsoumakas

14:15–14:30 Coffee break

14:30–16:00 Session 4: Posters

14:30–16:00 *Logical Layout Analysis Applied to Historical Newspapers*
Nicolas Gutehrlé and Iana Atanassova

14:30–16:00 *"Don't worry, it's just noise'": quantifying the impact of files treated as single textual units when they are really collections*
Thibault Clérice

14:30–16:00 *NLP in the DH pipeline: Transfer-learning to a Chronolect*
Aynat Rubinstein and Avi Shmidman

14:30–16:00 *Using Computational Grounded Theory to Understand Tutors' Experiences in the Gig Economy*
Lama Alqazlan, Rob Procter and Michael Castelle

14:30–16:00 *Can Domain Pre-training Help Interdisciplinary Researchers from Data Annotation Poverty? A Case Study of Legal Argument Mining with BERT-based Transformers*
Gechuan Zhang, David Lillis and Paul Nulty

Sunday, December 19, 2021 (continued)

14:30–16:00 *Japanese Beauty Marketing on Social Media: Critical Discourse Analysis Meets NLP*
Emily Öhman and Amy Gracy Metcalfe

14:30–16:00 *Text Zoning of Theater Reviews: How Different are Journalistic from Blogger Reviews?*
Mylene Maignant, Thierry Poibeau and Gaëtan Brison

14:30–16:00 *Word Sense Induction with Attentive Context Clustering*
Moshe Stekel, Amos Azaria and Shai Gordin

14:30–16:00 *Transferring Modern Named Entity Recognition to the Historical Domain: How to Take the Step?*
Baptiste Blouin, Benoit Favre, Jeremy Auguste and Christian Henriot

14:30–16:00 *TFW2V: An Enhanced Document Similarity Method for the Morphologically Rich Finnish Language*
Quan Duong, Mika Hämäläinen and Khalid Alnajjar

14:30–16:00 *Did You Enjoy the Last Supper? An Experimental Study on Cross-Domain NER Models for the Art Domain*
Alejandro Sierra-Múnera and Ralf Krestel

14:30–16:00 *An Exploratory Study on Temporally Evolving Discussion around Covid-19 using Diachronic Word Embeddings*
Avinash Tulasi, Asanobu Kitamoto, Ponnurangam Kumaraguru and Arun Balaji Buduru

Sentiment Dynamics of Success: Fractal Scaling of Story Arcs Predicts Reader Preferences

Yuri Bizzoni au701203@uni.au.dk
Telma Peura tpeura@cc.au.dk
Mads Rosendahl Thomsen madsrt@cc.au.dk
Kristoffer Nielbo kln@cas.au.dk

Abstract

We explore the correlation between the sentiment arcs of H. C. Andersen's fairy tales and their popularity, measured as their average score on the platform GoodReads. Specifically, we do not conceive a story's overall sentimental trend as predictive *per se*, but we focus on its coherence and predictability over time as represented by the arc's Hurst exponent. We find that degrading Hurst values tend to imply degrading quality scores, while a Hurst exponent between .55 and .65 might indicate a "sweet spot" for literary appreciation.

1 Introduction and Related Work

What determines the perception of **literary quality**? It can be argued that the feelings expressed by a story play a major role in its experience and assessment, since they activate similar **sentiments** in the readers' memory (Drobot, 2013; Hu et al., 2021b). At the same time, the relationship between a reader and the sentiments expressed by a text is anything but linear: the reader's perception of the sentiments on page depends on their context (the victory of a villain can engender unpleasant feelings, Maks and Vossen (2013)), and mechanisms like irony and catharsis can transform the negative feelings expressed by a text into positive feelings for a reader (Bosco et al., 2013). Ultimately, the intensity and nature of feelings on page does not tell us much about whether a reader will love, hate or remain indifferent towards a story. While it is of great interest to explore the temporal dimension of sentiments in literary texts (Gao et al., 2016; Cambria et al., 2017), and specific tools have been developed to this aim (Brooke et al., 2015; Jockers, 2017), it might not be the explicit sentimental trend of a story which plays a role in its quality, but the distribution and relative change of such sentiments through the story, independently from whether the direction of the narrative's arc is closer to a comedy or to a tragedy. We use nonlinear adapting filtering and **fractal analysis** to assess the inner **coherence** of a story arc and the auto-correlation in the sentiment dynamics at different time-scales, as measured through the Hurst exponent (Hu et al., 2021b). We then explore the correlation between such value and perceived literary quality.

We propose **H. C. Andersen**'s fairy tales as an ideal test case for our hypothesis: Andersen's tales tend to have relatively simple narratives, making it easier for us to explain their arcs and their level of predictability; they are well known and widely read in contemporary English, allowing us to retrieve the average rating from many readers (in the most popular cases, tens of thousands of readers); and to use state-of-the-art sentiment analysis tools. Lastly, some stories are short, which is a disadvantage when producing sentiment arcs, but is also a significant advantage when interpreting our results since we can re-read full stories in a short time and be assured of the quality of the computed arcs, making it much easier to explore the underlying causes of sentimental curves and their resulting Hurst exponents. While deciding whether a story's segment is happy or sad is a complex matter of interpretation, we can measure the sentimental value of the components from which such interpretations derive: the average positive or negative value of its words or sentences.

During the last two decades, **sentiment analysis** has generated a large number of resources to infer the kind and intensity of the sentiment expressed by a text, at the word (Mohammad and Turney, 2013), phrase (Agarwal et al., 2009; Hutto and Gilbert, 2014), sentence (Hu and Liu, 2001) or overall text level (Pang and Lee, 2004). Word-level resources, which are mainly constituted of manually or semi-automatically annotated **lexica** (Taboada et al., 2011), are popular in literary senti-

Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH), pages 1–6
December 19, 2021.

ment analysis (Gao et al., 2016). They provide the highest number of data points for each story, and they keep the inference of the arcs at the simplest possible level, requiring neither rule-based nor pre-trained scoring systems. While this is a limitation in the endeavour of inferring the sentiment of a portion of text, since sentiment lexica do not disambiguate the sense of a word in context (Zhang and Liu, 2017) and the analysis is limited to the words present in the used lexicon, this approach allows for drawing the succession of the smallest sentimental units of a text, from which human readers themselves draw their inferences. Furthermore, using lexica allows for drawing arcs in a transparent way, making it possible for researchers to immediately understand the causes of each score derived from a corpus. Finally, we use word-level sentiment arcs and their large-scale human ratings to determine whether the dynamic sentiment evolution and the level of coherence of a story arc at the sentiment level are connected with the perceived quality of the stories.

2 Data

Andersen corpus Our corpus consists of a collection of 126 H. C. Andersen's fairy tales translated into English retrieved from Project Gutenberg[1]. Their length varies between 1956 characters (*The Princess and the Pea*) and 106496 characters (*The Ice Maiden*), with an average length of 1585 characters.

SA lexicon To create the stories' sentiment arcs, we rely on the NRC-VAD lexicon (Mohammad, 2018), which is composed of almost 20.000 English words annotated for valence, arousal and dominance [2]. For this study, we only used valence.

GoodReads scores To measure the stories' perceived quality we resorted to GoodReads (Thelwall and Kousha, 2017), a popular web platform used to grade, comment and recommend books. We manually collected the average rating and the number of individual raters for each of Andersen's fairy tales (see Figure 1).

3 Methods

We proceed in three main steps: (i) we convert the stories into raw sentimental arcs through the use of the NRC-VAD lexicon (Section 3.1); (ii) we compute the story arcs' inner coherence through their Hurst coefficient (Section 3.2); (iii) we correlate each story's Hurst coefficient with its average rating on GoodReads (Section 4).

3.1 Arc Extraction

By retrieving the sentiment value of each word in our lexicon, we created a fine-grained sentiment arc for each story in the dataset. Since the lexicon we used only contains words with a non-neutral value, we assigned all out-of-vocabulary words a neutral value by default, in order to represent not only the highs and lows but the neutral sequences of the stories as well. After creating the arcs, we applied a smoothing technique to both improve their visualization and facilitate their detrending in post processing (see Figure 2). To control for the sensibility of our arcs, we selected a subset of 12 popular Andersen stories and grouped them in hierarchical sets through agglomerative clustering (Murtagh and Legendre, 2014). Two of the authors then independently checked the main clusters, verifying that they represented major, generally coherent sentimental groups.

3.2 Arc Coherence

Following Hu et. al (Hu et al., 2021a,b), we use the Hurst exponent to approximate the story arc's inner coherence. The Hurst exponent, H, is a measure of self-similar behavior. In the context of story arcs, self-similarity means that the arc's fluctuation patterns at faster time-scales resemble fluctuation patterns at slower time scales (Riley et al., 2012). We use Adaptive Fractal Analysis (AFA) to estimate the Hurst exponent (Gao et al., 2011).

AFA is based on a nonlinear adaptive multi-scale decomposition algorithm (Gao et al., 2011). The first step of the algorithm involves partitioning an arbitrary time series under study into overlapping segments of length $w = 2n + 1$, where neighboring segments overlap by $n + 1$ points. In each segment, the time series is fitted with the best polynomial of order M, obtained by using the standard least-squares regression; the fitted polynomials in overlapped regions are then combined to yield a single global smooth trend. Denoting the fitted polynomials for the $i - th$ and $(i + 1) - th$ segments by $y^i(l_1)$ and $y^{(i+1)}(l_2)$, respectively, where $l_1, l_2 = 1, \cdots, 2n + 1$, we define the fitting for the

[1] https://www.gutenberg.org/ebooks/27200

[2] We also experimented with the Vader lexicon (Hutto and Gilbert, 2014), but found its coverage too small to yield meaningful results at the word level.

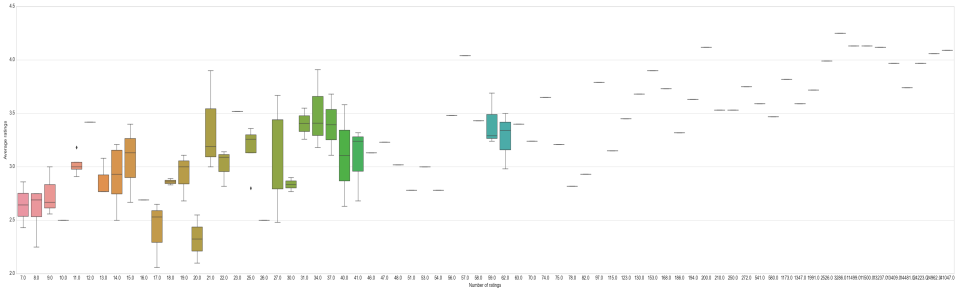

Figure 1: Number and magnitude of ratings. Most stories have less than 100 ratings, but the most known tales reach up to 40 thousand individual scores; stories with more raters tend to also receive higher average scores.

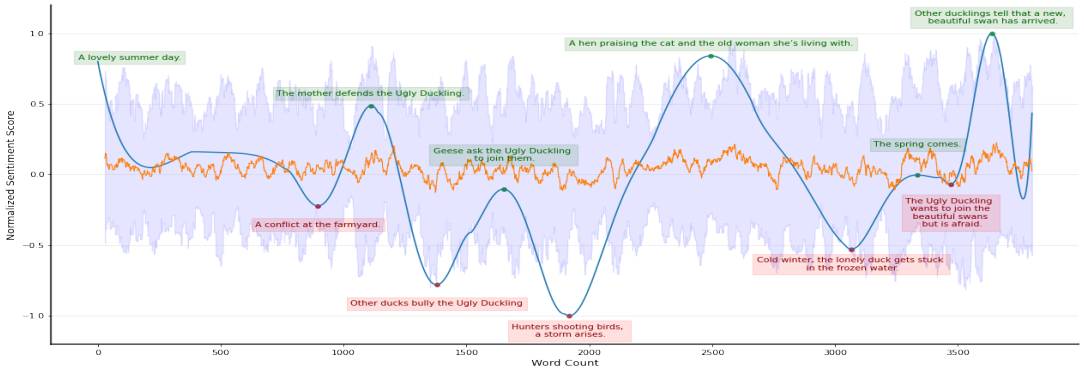

Figure 2: Sentiment arc from *The Ugly Duckling* with its narrative references. In orange and lilac, the mean and standard deviation of sentiment scores for every 30 words; the blue line is the polynomial smoothing of the raw sentiment arc.

overlapped region as

$$y^{(c)}(l) = w_1 y^{(i)}(l+n) + w_2 y^{(i+1)}(l), \quad l = 1, 2, \cdots, n+1$$

where $w_1 = \left(1 - \frac{l-1}{n}\right)$ and $w_2 = \frac{l-1}{n}$ can be written as $(1 - d_j/n)$ for $j = 1, 2$, and where d_j denotes the distances between the point and the centers of $y^{(i)}$ and $y^{(i+1)}$, respectively. Note that the weights decrease linearly with the distance between the point and the center of the segment. Such a weighting is used to ensure symmetry and effectively eliminate any jumps or discontinuities around the boundaries of neighboring segments. As a result, the global trend is smooth at the non-boundary points, and has the right and left derivatives at the boundary (Riley et al., 2012). The global trend thus determined can be used to maximally suppress the effect of complex nonlinear trends on the scaling analysis. The parameters of each local fit is determined by maximizing the goodness of fit in each segment. The different polynomials in overlapped parts of each segment are combined so that the global fit will be the best (smoothest) fit of the overall time series. Note that, even if $M = 1$ is selected, i.e., the local fits are linear, the global trend signal will still be nonlinear. With the above procedure, AFA can be readily described. For an arbitrary window size w, we determine, for the random walk process $u(i)$, a global trend $v(i), i = 1, 2, \cdots, N$, where N is the length of the walk. The residual of the fit, $u(i) - v(i)$, characterizes fluctuations around the global trend, and its variance yields the Hurst parameter H according to the following scaling equation:

$$F(w) = \left[\frac{1}{N}\sum_{i=1}^{N}(u(i) - v(i))^2\right]^{1/2} \sim w^H$$

By computing the global fits, the residual, and the variance between original random walk process and the fitted trend for each window size w, we can plot $\log_2 F(w)$ as a function of $\log_2 w$. The presence of fractal scaling amounts to a linear relation in the plot, with the slope of the relation providing an estimate of H[3]. Accordingly, a H higher

[3]Code for computing DFA and AFA is available at https://github.com/knielbo/saffine.

than .5 indicates a degree of linear coherence (e.g., positive sentiments are followed by positive sentiments), while H lower than .5 indicates a series that tends to revert to the mean (e.g. a positive emotion always follows a negative emotion). Stories based on the repetition of the same narrative mechanism like $The Butterfly$, where a butterfly meets several nice flowers but always finds them faulty, have a relatively low Hurst exponent (see Figure 4).

4 Results

Our final step consisted in correlating each fairy tale's average rating with its average Hurst coefficient. We found that there is a small, but significant correlation between the Hurst exponent of Andersen's tales and their average success on Good Reads (see Figure 3).

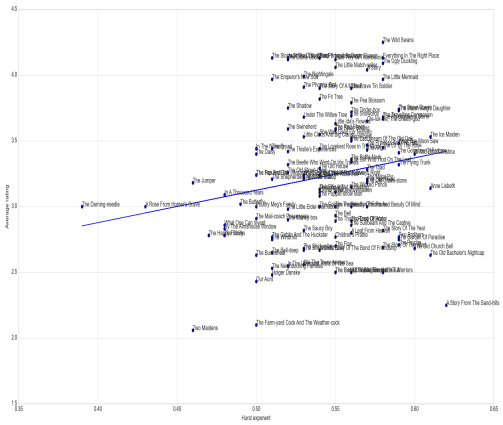

Figure 3: Correlation between Hurst exponent and average rating for all tales.

Nonetheless, there appears to be a difference in such correlation when we consider the number of ratings each story received. As we discussed before, the statistical value of those ratings can change: obscure stories can have less than ten different raters, while the most known fairy tales can be rated by tens of thousands of readers. Many of Andersen's stories are not well known to the larger public, and thus received a relatively small number of reviews. Based on the average number of ratings of each story, we recomputed the correlation strength for only those tales that received more than 30 different scores, which excluded around half of our corpus, leaving 63 tales in the corpus. This threshold allowed us to keep only stories that had a significant amount of individual annotations, without excessively reducing the size of our dataset. The result was a much stronger correlation, and a more significant p-value (see Figure 4). In Table 1, we show a summary of the correlation values we obtained on both sets. Finally, in Figure 5 we draw the overall intuition of the study: works with a smaller Hurst exponent feature mean-reverting sentiment trends and elicit lower overall appreciation.

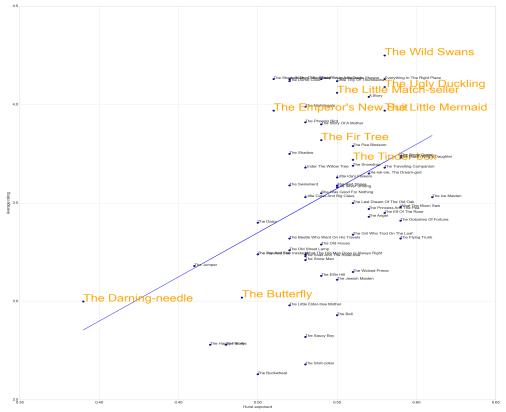

Figure 4: Correlation between Hurst exponent and average rating for tales having more than 30 ratings with some titles in evidence. Most of the very popular stories fall on the upper right corner, while tales like *The Butterfly*, based on a simple repetition of the same dynamic, fall more to the left. Also notice how without the outlier *The Darning Needle*, the correlation of the remaining data points would be even steeper.

	All tales		**Popular tales**	
	corr.	p value	corr.	p value
Pearson	.19	.03	.4	.001
Spearmann	.18	.04	.35	.005
Kendall Tau	.12	.03	.23	.009
Distance corr.	.81		.6	

Table 1: Correlations for all tales and tales with more than 30 ratings ("popular"). Statistical significance is not directly applicable to standard distance correlation.

5 Conclusions and Future Work

The main finding of this paper is that there is a correlation between a story's sentimental coherence and its perceived quality. With all the necessary caveats, we find that such correlation is an interesting result and advocates for a more extensive use of multifractal theory in the study of sentimental arcs

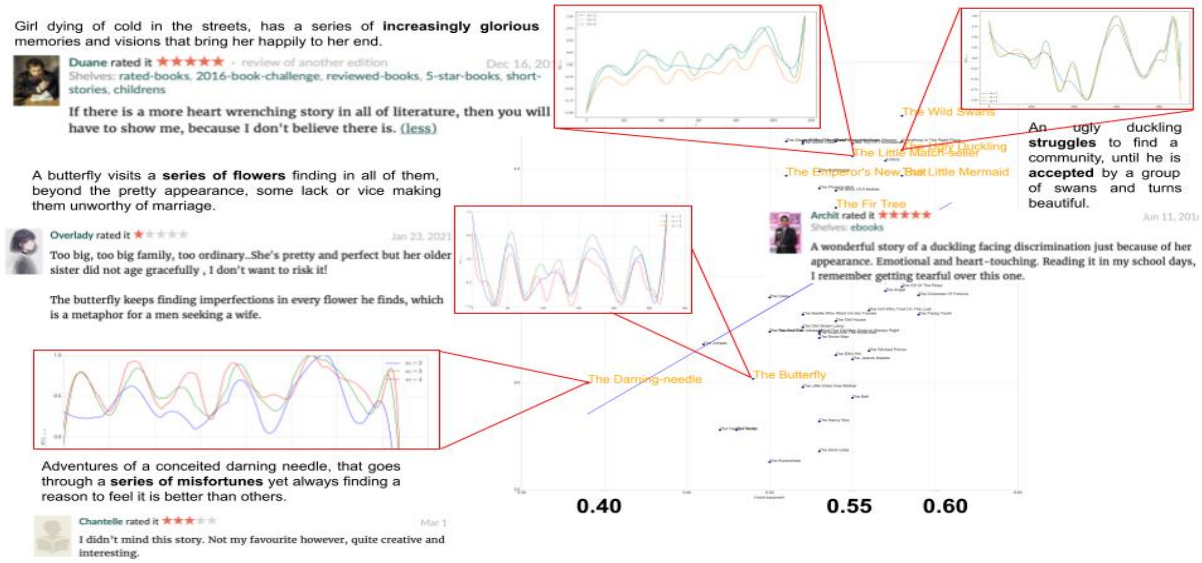

Figure 5: A visual summary of our concept. More "zig-zag" lines reverting to the mean tend to receive less favorable overall reviews than stories having a smoother trendline. The latter ones also include many of the most known Andersen's stories.

in literature. The emotional coherence of a story, as represented by the average Hurst exponent of the sentiment arc, explains a part of its perceived quality as measured by its average rating on a general audience website. Furthermore, stories with more ratings tend to show a stronger correlation between these two measures, which might mean that the weaker correlations we recorded for the less known stories might be due to an insufficiently large pool of raters (thus having a less robust score). It is interesting to notice that stories with many ratings also tend to have higher ratings on average: this further shows how, at least for fairy tales, fame and popularity tend to go together. While it proved surprisingly predictive, reducing a story arc to one overall Hurst coefficient means losing important information in terms of how coherence is distributed through the narrative. Exploring the temporal variation of this coefficient at different time scales of a story might reveal further insights into the sentiment dynamics of a good narrative evolution. Another aspect for further exploration is evaluating alternative ways of scoring the sentiments in a text. Particularly, when applied to longer stories, adopting a sentence-level approach could help accounting for the context that indeed affects the sentiment interpretation. Furthermore, instead of sentiment analysis, moving to an emotion analysis might allow for more detailed insights about an optimal narrative development. However, already the present results suggest that there exists a desirable ratio of coherence and unpredictability in the sentiment arcs that contribute to the appreciation of a story.

Acknowledgments

This paper has been supported the 'Fabula-NET: A Deep Neural Network for Automated Multidimensional Assessment of Literary Fiction and Narratives' funded by the Velux Foundation, and the DeiC Interactive HPC system with project DeiC-AU1-L-000015.

References

Apoorv Agarwal, Fadi Biadsy, and Kathleen Mckeown. 2009. Contextual phrase-level polarity analysis using lexical affect scoring and syntactic n-grams. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 24–32.

Cristina Bosco, Viviana Patti, and Andrea Bolioli. 2013. Developing corpora for sentiment analysis: The case of irony and senti-tut. *IEEE intelligent systems*, 28(2):55–63.

Julian Brooke, Adam Hammond, and Graeme Hirst. 2015. Gutentag: an nlp-driven tool for digital humanities research in the project gutenberg corpus. In *Proceedings of the Fourth Workshop on Computational Linguistics for Literature*, pages 42–47.

Erik Cambria, Dipankar Das, Sivaji Bandyopadhyay, and Antonio Feraco. 2017. Affective computing and sentiment analysis. In *A practical guide to sentiment analysis*, pages 1–10. Springer.

Irina-Ana Drobot. 2013. Affective narratology. the emotional structure of stories. *Philologica Jassyensia*, 9(2):338.

Jianbo Gao, Jing Hu, and Wen-wen Tung. 2011. Facilitating Joint Chaos and Fractal Analysis of Biosignals through Nonlinear Adaptive Filtering. *PLoS ONE*, 6(9):e24331.

Jianbo Gao, Matthew L Jockers, John Laudun, and Timothy Tangherlini. 2016. A multiscale theory for the dynamical evolution of sentiment in novels. In *2016 International Conference on Behavioral, Economic and Socio-cultural Computing (BESC)*, pages 1–4. IEEE.

M Hu and B Liu. 2001. Mining and summarizing customer reviews. in (kohavi, r., hrsg.): Proceedings of the 2004 acm sigkdd international conference on knowledge discovery and data mining, seattle, 2004. acm [jb01] jacquemin, c.; bourigault, d.: Term extraction and automatic indexing. in (mitkov, r. hrsg.): Handbook of computational linguistics. *Oxford University Press*, 10:1014052–1014073.

Qiyue Hu, Bin Liu, Jianbo Gao, Kristoffer L Nielbo, and Mads Rosendahl Thomsen. 2021a. Fractal scaling laws for the dynamic evolution of sentiments in never let me go and their implications for writing, adaptation and reading of novels. *World Wide Web*, pages 1–18.

Qiyue Hu, Bin Liu, Mads Rosendahl Thomsen, Jianbo Gao, and Kristoffer L Nielbo. 2021b. Dynamic evolution of sentiments in never let me go: Insights from multifractal theory and its implications for literary analysis. *Digital Scholarship in the Humanities*, 36(2):322–332.

Clayton Hutto and Eric Gilbert. 2014. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 8.

Matthew Jockers. 2017. Syuzhet: Extracts sentiment and sentiment-derived plot arcs from text (version 1.0. 1).

Isa Maks and Piek Vossen. 2013. Sentiment analysis of reviews: Should we analyze writer intentions or reader perceptions? In *Proceedings of the International Conference Recent Advances in Natural Language Processing RANLP 2013*, pages 415–419.

Saif M. Mohammad. 2018. Obtaining reliable human ratings of valence, arousal, and dominance for 20,000 english words. In *Proceedings of The Annual Conference of the Association for Computational Linguistics (ACL)*, Melbourne, Australia.

Saif M Mohammad and Peter D Turney. 2013. Nrc emotion lexicon. *National Research Council, Canada*, 2.

Fionn Murtagh and Pierre Legendre. 2014. Ward's hierarchical agglomerative clustering method: which algorithms implement ward's criterion? *Journal of classification*, 31(3):274–295.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 271–278.

Michael A. Riley, Scott Bonnette, Nikita Kuznetsov, Sebastian Wallot, and Jianbo Gao. 2012. A tutorial introduction to adaptive fractal analysis. *Frontiers in Physiology*, 3.

Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. 2011. Lexicon-based methods for sentiment analysis. *Computational linguistics*, 37(2):267–307.

Mike Thelwall and Kayvan Kousha. 2017. Goodreads: A social network site for book readers. *Journal of the Association for Information Science and Technology*, 68(4):972–983.

Lei Zhang and Bing Liu. 2017. Sentiment analysis and opinion mining.

The Validity of Lexicon-based Emotion Analysis in Interdisciplinary Research

Emily Öhman
Waseda University
ohman@waseda.jp

Abstract

Lexicon-based sentiment and emotion analysis methods are widely used particularly in applied Natural Language Processing (NLP) projects in fields such as computational social science and digital humanities. These lexicon-based methods have often been criticized for their lack of validation and accuracy – sometimes fairly. However, in this paper, we argue that lexicon-based methods work well particularly when moving up in granularity and show how useful lexicon-based methods can be for projects where neither qualitative analysis nor a machine learning-based approach is possible. Indeed, we argue that the measure of a lexicon's accuracy should be grounded in its usefulness.

1 Introduction

Lexicon-based sentiment analysis is probably the simplest approach to determining the polarity or emotional content of a text. At the core, it is simply comparing the lemmatized tokens in a text being analyzed to the lemmas in the lexicon and assigning them a score accordingly. Some models go a step further and try to include valence-shifters in the final polarity measure, but most commonly lexicon-based sentiment analysis relies on bag-of-words approaches (see e.g. Taboada et al. (2011)).

Lexicon-based methods are commonly contrasted with machine learning-based methods (Nguyen et al., 2018; Kaushik and Mishra, 2014; van Atteveldt et al., 2021). Machine learning is typically context sensitive and can be combined with large language models for a fairly accurate picture of the linguistic content of a text. Machine learning models typically perform much better than lexicon-based models on sentiment analysis tasks when comparing traditional evaluation metrics (Kaushik and Mishra, 2014; González-Bailón and Paltoglou, 2015; Dhaoui et al., 2017; van Atteveldt et al., 2021).

However, there are two issues with comparing machine learning, or data-driven, and lexicon-based models. The first is that an accurate machine learning model needs labeled data for training, validation, and testing. Labeling or annotating data generally requires at least three human annotators who need to be compensated for their work. Therefore machine learning datasets can quickly exceed the budget of many projects or be flat out impossible to conduct properly especially for early career researchers with smaller amounts of grant money to spend on such tasks (see e.g. Gatti et al. (2015)).

The second issue, which is rarely discussed, is that these evaluation metrics are not really comparable due to how emotion and sentiment scores are assigned and calculated using these different methods. Naturally, any approach needs to be evaluated, but in practice it is much harder to accurately evaluate the output of a lexicon-based model. Instead, the focus should be on usefulness of the output accompanied by a sanity check of the results. The validation issue is discussed in detail in section 3.

It is rare to see lexicon-based methods used for sentiment analysis in NLP papers. Conversely, it is fairly common to see them used in interdisciplinary projects. In some fields, there are very few scholars willing to review interdisciplinary papers, and even fewer who have the expertise to properly make judgments on the methodology. This can lead to some papers being accepted that have dubious methodology or other being rejected because they are too technical. This is something that affects interdisciplinary fields the most.

In the following pages we present an overview of interdisciplinary sentiment analysis practices and common criticism against different methods. We also discuss the issue with the evaluation of lexicon-based sentiment analysis projects and offer

Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH), pages 7–12
December 19, 2021.

some preliminary solutions while making a case for lexicon-based sentiment analysis in interdisciplinary projects.

2 Background

2.1 The Creation and Validation of Emotion Lexicons

There are a few different ways of creating emotion lexicons, however, typically some type of emotion dictionary is used to extract relevant lexical items (Mohammad and Turney, 2010). There are many ways of annotating for emotions. Annotators might be asked to annotate for emotion evocation or emotion association, which can result in very different results. Mohammad and Turney (2013) found that annotating for emotion association resulted in more reliable annotations. Annotator fatigue is also very common, especially when annotating for emotions or sentiments (Mohammad, 2016; Öhman, 2020a) and therefore the method of annotation also has a significant impact on the quality of annotations (Kiritchenko and Mohammad, 2017).

Nonetheless, these lexicons are carefully constructed and inter-annotator agreement scores are carefully evaluated. Noisy annotations and even noisy annotators are often removed before compiling the final lexicon. The reality is simply that human annotators do not always agree on an annotation, and when the annotation task is emotion annotation, disagreements are even more common (Strapparava and Mihalcea, 2007; Andreevskaia and Bergler, 2007; Wiebe and Riloff, 2005). A typical inter-annotator agreement percentage is around 70% but can be much lower than that (Bermingham and Smeaton, 2009; Ng et al., 1999).

2.2 Common complaints

Particularly in computational social sciences and digital humanities the use of black-box or black-box-like tools is fairly common (Lazer et al., 2020; Gefen et al., 2021). Often this tool is LIWC (Linguistic Inquiry and Word Count) (Pennebaker et al., 2001) (especially in Computational Social Sciences) (Puschmann and Powell, 2018). In essence LIWC is an emotion lexicon and part-of-speech tagger (with other additional features).

LIWC itself has attracted criticism beyond the typical complaints against lexicon-based methods (Puschmann and Powell, 2018), not because the lexicon is any worse than any other emotion lexicon, but because the creators of LIWC not only claim that LIWC can detect emotions in text, but that it can also accurately identify a person's psychological state (Tausczik and Pennebaker, 2010). Indeed, this approach has been attempted in digital humanities research too (Boyd, 2017).

Another famous example is the Syuzhet package for R (Jockers, 2015). In essence, Syuzhet accepts a text (a novel) as input, compares the words in the text to those in the lexicons available to it, and outputs different visualizations of sentiment polarities in the text along the narrative path. Syuzhet has received multiple complaints ranging from statistical issues to the very practical issue of valence shifters not being taken into consideration and an over-reliance of word-occurrence (Swafford, 2015, 2016).

In their excellent work Data-sitter's club, DH project of the year 2019, Bowers and Dombrowski (2019) exemplify the problem with sentiment analysis in digital humanities by comparing some common programs for sentiment analysis such as Vader-Sentiment (Hutto and Gilbert, 2014) and TextBlob (Loria, 2018). They look at individual sentences and compare their human judgment with the judgments of these programs that are lexicon-based. Their conclusion is that sentiment analysis, particularly lexicon-based, is highly inaccurate.

These examples highlight the issue with lexicon-based approaches for emotion detection. Even the most accurate emotion lexicon still just counts words in the target text. To make claims beyond the occurrence of emotion-associated words is misguided at best and disingenuous at worst. However, this does not mean that there is something wrong with the lexicons themselves. It also does not mean that these lexicons can not be useful for sentiment and emotion analysis.

2.3 Comparison between lexicon-based and data-driven approaches

González-Bailón and Paltoglou (2015) compared the performance of several off-the-shelf sentiment lexicons to a machine learning approach. They concluded that lexicon-based methods performed comparable to machine learning, but that the accuracy of lexicon-based methods suffered more when the content was more diverse or informal. They calculated the accuracy by comparing to human annotators. Their final recommendation is to use machine learning.

van Atteveldt et al. (2021) do something similar,

but they also add a layer of complication to their validity measures in that they translate their content to English from Dutch in order to use many off-the-shelf lexicons. Nonetheless, their results too, when compared to human coders, suggest that machine learning approaches have a much higher accuracy than lexicon-based ones.

3 The Issue with Validation

Validation is a complex matter, especially when we are evaluating lexicon-based emotion analysis projects in digital humanities projects that analyze more than the performance of a model. The approaches to validation differ greatly between fields. Some simply trust that LIWC does what it says it does (as evidenced by over 13,000 combined citations for the tool on Google Scholar[1]) and complete no further validation or sanity check, some look at a few individual data points and annotate or evaluate this themselves by comparing directly with the results from their model's output.

Naturally, for a lexicon-based method to be considered useful, the output should be close to a qualitative evaluation by a human. However, as already discussed, humans rarely agree on the emotional content of a word, sentence, or paragraph, so whether we are using annotated data for lexicons or to train machine learning models, it is hardly surprising that the output mirrors the confusion in the data. Only with highly disjoint categories and quality annotations devoid of noise is it possible to get high accuracies with classification tasks (see e.g. Demszky et al. (2020) and Abdul-Mageed and Ungar (2017)).

van Atteveldt et al. (2021) suggests that to measure the validity of a lexicon-based approach, one should manually annotate at least 100 data points, but ideally 300 for accurate Krippendorf's α scores. But if we are examining literary works or political party manifestos, this is not really possible as the unit of evaluation is typically a full document and we rely on composite scores for a unit at a coarser granularity than what the model is evaluating at. Furthermore, if a lexicon offers a range of scores beyond 0 and 1, such as intensity scores between 0 and 1 for each lexical item it is quite difficult to source these human annotations as humans are notoriously bad at rating scales and the emotion intensities in the NRC Emotion Intensity lexicon (Mohammad et al., 2018), for example, were obtained using best-worst-scaling (Kiritchenko and Mohammad, 2017), something which is typically not feasible to conduct for small batches of test data.

The problem with the suggested validation steps of such results is that (1) they work best for evaluating binary or ternary **sentiment** categorization, and (2) they evaluate computational approaches against human annotations. In some cases the latter makes sense. If we are analyzing tweets or other short messages for sentiment or emotion it makes sense to look at the assigned emotion scores at sentence- or message-level and these are relatively easy to compare against human annotations. However, the manual annotation for emotions typically produces different output than what lexicon-based models do and direct comparisons can be difficult in the best of circumstances. If we are working with emotion analysis with six, eight, or even more categories or emotion intensities instead of binary categories the results become even more complex and more difficult to compare against human annotations or machine learning approaches (Öhman, 2020b). The expectation still seems to be to follow the guidelines of binary sentiment analysis validation at sentence-level even when using multiple emotion categories for emotion intensity at document level.

If we are analyzing the emotional intensity of each named emotion in the content of speeches, party manifestos, or romance novels we are typically doing this analysis for chunks of 3,000-10,000 words. Following the validation guidelines of van Atteveldt et al. (2021), i.e. annotating a minimum of 100 units would mean manually annotating the emotional intensity of at least 300,000 words by 2-3 annotators. This amount of annotations is not even necessary to train a machine learning model. The next best thing then becomes annotating 2-3 chunks at sentence-level and calculating a composite score of the human annotations as well. This would typically result in at least 1,000 manual annotations which is more than enough to calculate Krippendorf's α accurately. However, this still leaves us with the issue of how human annotators would reliably be able to annotate for emotional intensity as there is little **intra**-annotator agreement, let alone inter-annotator agreement when annotating for scale. Furthermore, most lexicon-based models would likely score a sentence with two words expressing *sadness* as having twice the

[1] Accessed on November 12, 2021

sadness of a sentence that contained only one such word, but when a human annotator manually annotated that sentence, it is quite likely that they would only mark it as containing *sadness* in general, again making direct comparisons more difficult.

4 Proposed Solution & Use case

The first step would be to stop calling what lexicon-based methods do *emotion* or *sentiment analysis* and refer to it as analyzing the distribution of emotion-laden words. This is a much more accurate description of what lexicon-based methods actually do, especially when contrasted with what machine learning based methods do. If lexicon-based approaches are used together with statistical significance calculations, we can show that there are significant differences between the use of words associated with specific emotions in two comparable texts. This is in itself a demonstration of usefulness.

Such an approach also minimizes the need for adjusting the results for valence-shifters. If we are evaluating the use of emotion words in novels, whether an emotion word is negated or not is not as relevant because in this case authors choose their words to evoke specific emotions in the reader and thus such an approach is excellent for measuring tone and mood in text. It might even be argued that such an approach to tone and mood is going to result in more relevant results than a machine learning approach would as it might be easier to access the author's intent rather than the surface of the words. Word choice by literary authors has been shown to affect the mood of the novel significantly (McCormack, 2006; Ngai, 2005).

Another domain where words are carefully chosen is politics (Riggins, 1997; Orwell, 1946). Comparing the content of two political manifestos the distribution of emotion words when combined with statistical significance testing, can show us what type of emotion words are used more in each of the manifestos, and whether the difference is statistically significant. If the differences are statistically significant, the fact that the results indicate that one party used different words to evoke different emotions or words of different intensity is a useful finding. As a side note, especially when using off-the-shelf general purpose lexicons, it is a good idea to stick to formal single-domain texts in order to maximize the validity of the results as suggested by the results of González-Bailón and Paltoglou (2015).

The solution is to establish an evaluation metric for lexicon-based methods that focuses on usefulness rather than accuracy. A part of this usefulness measure would include doing some type of sanity check or validation comparing to human impressions of the text, but would take into account the different outputs of the lexicon-based model and the human annotations. A part of this validation can indeed use Krippendorf's α scores to check for inter-annotator agreement between the human annotators, as these annotators would have annotated the text in comparable ways. The comparison between the outputs of the model and the human annotators requires other metrics to determine usefulness or even traditional accuracy depending on what exactly the model outputs.

4.1 Use case

We have achieved the best results by letting the model add word scores that are then combined at document-level for a document emotion-word intensity score for each emotion using Plutchik's 8 core emotions sans *surprise* (Plutchik, 1980) as *surprise* is notoriously difficult to detect in text, particularly at sentence-level or finer granularity (Alm and Sproat, 2005). The human annotators (at least 2, but ideally 3) annotate a few select representative documents at roughly sentence level by simply marking the sentence as containing the emotions in the annotation scheme. Although humans annotate for the binary existence or non-existence of the particular emotions, the results are far more reliable than if they were to annotate for intensity (Kiritchenko and Mohammad, 2017). The results also correlate highly with those of the intensity-scores both in terms of absolute numbers and proportional distribution of emotions.

In one instance we examined Finnish political party manifestos (Koljonen et al., forthcoming). We used a straight-forward emotion intensity lexicon that had been adjusted for political data and the Finnish language (Öhman, forthcoming) to get composite scores for nearly 1000 party manifestos that were on average around 20,000 tokens in length. Using linear regression to analyze statistical significance showed that the main difference between different parties, manifesto types, and eras, was that although populist parties used the same amount of emotion words as other parties, the words they used were of significantly higher

intensity.

We confirmed our findings by having three annotators annotate three manifestos, by different political parties and different eras, manually at approximately sentence-level by marking that sentence as expressing or not expressing any of the emotions in our scheme. We calculated inter-annotator agreement using Krippendorf's α which was on par with other emotion annotation tasks. We then compared that score to the compound score adjusted for word count from our model. The values per emotion were nearly identical. It was not possible to do a direct inter-rater agreement calculation, but comparing the distribution of emotions, the values were again nearly identical for all the target manifestos.

Comparing the manual annotations to the output of the lexicon would not have yielded any useful metrics. However, the significance calculations show that there was valuable undiscovered information in the data that we could access with emotion lexicons.

5 Concluding Discussion

In this opinion paper we have tried to justify the use of lexicon-based emotion analysis, particularly in interdisciplinary research. There is little doubt that data-driven methods such as machine learning are typically the best choice when aiming for accuracy, however, there are projects and approaches where lexicon-based methods fare equally well, and sometimes are even more suitable for the task than machine learning. We hope this paper initiates a discussion in particular about the process of validating results from lexicon-based approaches in a way that would recognize the usefulness of lexicon-based approaches for specific types of text commonly used in digital humanities.

References

Muhammad Abdul-Mageed and Lyle Ungar. 2017. Emonet: Fine-grained emotion detection with gated recurrent neural networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 718–728.

Cecilia Ovesdotter Alm and Richard Sproat. 2005. Emotional sequencing and development in fairy tales. In *International Conference on Affective Computing and Intelligent Interaction*, pages 668–674. Springer.

Alina Andreevskaia and Sabine Bergler. 2007. CLaC and CLaC-NB: Knowledge-based and corpus-based approaches to sentiment tagging. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, pages 117–120, Stroudsburg, PA, USA. Association for Computational Linguistics.

Wouter van Atteveldt, Mariken ACG van der Velden, and Mark Boukes. 2021. The validity of sentiment analysis: Comparing manual annotation, crowd-coding, dictionary approaches, and machine learning algorithms. *Communication Methods and Measures*, 15(2):121–140.

Adam Bermingham and Alan F Smeaton. 2009. A study of inter-annotator agreement for opinion retrieval. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 784–785.

Katherine Bowers and Quinn Dombrowski. 2019. Katia and the sentiment snobs. In *The Data-Sitters Club*.

Ryan L Boyd. 2017. Psychological text analysis in the digital humanities. In *Data analytics in digital humanities*, pages 161–189. Springer.

Dorottya Demszky, Dana Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, and Sujith Ravi. 2020. Goemotions: A dataset of fine-grained emotions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4040–4054.

Chedia Dhaoui, Cynthia M Webster, and Lay Peng Tan. 2017. Social media sentiment analysis: lexicon versus machine learning. *Journal of Consumer Marketing*.

Lorenzo Gatti, Marco Guerini, and Marco Turchi. 2015. Sentiwords: Deriving a high precision and high coverage lexicon for sentiment analysis. *IEEE Transactions on Affective Computing*, 7(4):409–421.

Alexandre Gefen, Léa Saint-Raymond, and Tommaso Venturini. 2021. Ai for digital humanities and computational social sciences. In *Reflections on Artificial Intelligence for Humanity*, pages 191–202. Springer.

Sandra González-Bailón and Georgios Paltoglou. 2015. Signals of public opinion in online communication: A comparison of methods and data sources. *The ANNALS of the American Academy of Political and Social Science*, 659(1):95–107.

Clayton J. Hutto and Eric Gilbert. 2014. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *ICWSM*. The AAAI Press.

Matthew L Jockers. 2015. Syuzhet: Extract sentiment and plot arcs from text. *blog post*.

Chetan Kaushik and Atul Mishra. 2014. A scalable, lexicon based technique for sentiment analysis. *arXiv preprint arXiv:1410.2265*.

Svetlana Kiritchenko and Saif Mohammad. 2017. Best-worst scaling more reliable than rating scales: A case study on sentiment intensity annotation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 465–470. Association for Computational Linguistics.

Juha Koljonen, Emily Öhman, Mikko Mattila, and Pertti Ahonen. forthcoming. Strength and intensity of sentiments and emotions in party manifestos: Finland, 1945 to 2019.

David MJ Lazer, Alex Pentland, Duncan J Watts, Sinan Aral, Susan Athey, Noshir Contractor, Deen Freelon, Sandra Gonzalez-Bailon, Gary King, Helen Margetts, et al. 2020. Computational social science: Obstacles and opportunities. *Science*, 369(6507):1060–1062.

Steven Loria. 2018. textblob documentation. *Release 0.15*, 2.

Thomas McCormack. 2006. *The fiction editor, the novel, and the novelist*. Paul Dry Books.

Saif Mohammad. 2016. A practical guide to sentiment annotation: Challenges and solutions. In *WASSA@ NAACL-HLT*, pages 174–179.

Saif Mohammad, Felipe Bravo-Marquez, Mohammad Salameh, and Svetlana Kiritchenko. 2018. SemEval-2018 task 1: Affect in tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 1–17. Association for Computational Linguistics.

Saif Mohammad and Peter Turney. 2010. Emotions evoked by common words and phrases: Using mechanical turk to create an emotion lexicon. In *Proceedings of the NAACL HLT 2010 workshop on computational approaches to analysis and generation of emotion in text*, pages 26–34.

Saif M. Mohammad and Peter D. Turney. 2013. Crowdsourcing a word-emotion association lexicon. *Computational Intelligence*, 29(3):436–465.

Hwee Tou Ng, Chung Yong Lim, and Shou King Foo. 1999. A case study on inter-annotator agreement for word sense disambiguation. In *SIGLEX99: Standardizing Lexical Resources*.

Sianne Ngai. 2005. *Ugly feelings*, volume 6. Harvard University Press Cambridge, MA.

Heidi Nguyen, Aravind Veluchamy, Mamadou Diop, and Rashed Iqbal. 2018. Comparative study of sentiment analysis with product reviews using machine learning and lexicon-based approaches. *SMU Data Science Review*, 1(4):7.

Emily Öhman. 2020a. Challenges in Annotation: Annotator Experiences from a Crowdsourced Emotion Annotation Task. In *Digital Humanities in the Nordic Countries 2020*. CEUR Workshop Proceedings.

Emily Öhman. 2020b. Emotion annotation: Rethinking emotion categorization. In *DHN Post-Proceedings*, pages 134–144.

Emily Öhman. forthcoming. SELF & FEIL: Sentiment and Emotion Lexicons for Finnish.

George Orwell. 1946. Politics and the english language.

James W Pennebaker, Martha E Francis, and Roger J Booth. 2001. Linguistic inquiry and word count: Liwc 2001. *Mahway: Lawrence Erlbaum Associates*, 71(2001):2001.

Robert Plutchik. 1980. A general psychoevolutionary theory of emotion. *Theories of emotion*, 1:3–31.

Cornelius Puschmann and Alison Powell. 2018. Turning words into consumer preferences: How sentiment analysis is framed in research and the news media. *Social Media+ Society*, 4(3):2056305118797724.

Stephen Harold Ed Riggins. 1997. *The language and politics of exclusion: Others in discourse.* Sage Publications, Inc.

Carlo Strapparava and Rada Mihalcea. 2007. Semeval-2007 task 14: Affective text. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 70–74.

Annie Swafford. 2015. Problems with the Syuzhet package. *Anglophile in Academia: Annie Swafford's Blog*.

Joanna Swafford. 2016. Messy data and faulty tools. *JSTOR*.

Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. 2011. Lexicon-based methods for sentiment analysis. *Computational linguistics*, 37(2):267–307.

Yla R Tausczik and James W Pennebaker. 2010. The psychological meaning of words: Liwc and computerized text analysis methods. *Journal of language and social psychology*, 29(1):24–54.

Janyce Wiebe and Ellen Riloff. 2005. Creating subjective and objective sentence classifiers from unannotated texts. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 486–497. Springer.

How Does the Hate Speech Corpus Concern Sociolinguistic Discussions? A Case Study on Korean Online News Comments

Won Ik Cho
Dept. of ECE and INMC
Seoul National University
tsatsuki@snu.ac.kr

Jihyung Moon
Spekt
jihyung.moon@spekt.to

Abstract

Social consensus has been established on the severity of online hate speech since it not only causes mental harm to the target, but also gives displeasure to the people who read it. For Korean, the definition and scope of hate speech have been discussed widely in researches, but such considerations were hardly extended to the construction of hate speech corpus. Therefore, we create a Korean online hate speech dataset with concrete annotation guideline to see how real world toxic expressions concern sociolinguistic discussions. This inductive observation reveals that hate speech in online news comments is mainly composed of social bias and toxicity. Furthermore, we check how the final corpus corresponds with the definition and scope of hate speech, and confirm that the overall procedure and outcome is in concurrence with the sociolinguistic discussions.

1 Introduction

Hate speech is an issue that has permeated deeply into our daily life (ElSherief et al., 2018). Hate speech is often explicitly stated along with insulting expressions, and some of them are perceived as hateful or offensive just by incorporating social bias. Accordingly, public figures or underrepresented groups suffer from tremendous mental damage, while some experience depression or end their lives.

Regarding the hate speech in online spaces, discussions are divided into *definition* and *detection* (MacAvaney et al., 2019) point of view. *Definition* mainly concerns "*What the hate speech is*" in a deductive manner, while *detection* attacks the issue with more an inductive methodology. For Korean, the hate speech study has mainly been conducted in the sociolinguistics community regarding the definition, and these include the discussion on the appropriateness of expression "hate speech" itself (Hong et al., 2016), its scope (Kim, 2017b), and the legal issues around discrimination and insult (Park and Choo, 2017). In Hong et al. (2016), hate speech is defined as "an expression that discriminates/hates or instigates discrimination/hostility/violence for some social minority individual/group". To back up these studies, further discussions on the underrepresentedness of each group have also actively taken place (Kim, 2017a, 2018).

However, aside from the importance of such discussions, there is a gap between the theoretical definition of *hate speech* and real *hateful expressions* that appear in our lives (Davidson et al., 2017). From a detection perspective, the following questions are mainly discussed which are not easy to answer in a definition point of view: "Should a certain expression be regarded as *hateful* even if a majority of people do not feel offensive for the same sentence?", "What if for the pre-existing terms that a small group of people insists on its harm?", and "How about the toxic expressions that head the criminals?". If there is a consensus on these issues, collecting data to develop a model for hate speech detection would be more clear.

Most previous approaches on Korean online hate speech detection have been keyword-based that regards glossaries on profanity terms[1] (Kang, 2018; Park and Cha, 2018). It is also challenging to find cases of constructing a corpus referring to preceding researches in other cultural regions (Waseem and Hovy, 2016; Davidson et al., 2017; Basile et al., 2019). Therefore, to understand how hate speech is represented in the Korean online expressions and how the inductive analysis corresponds with the concurrent discussions, we should investigate the attributes of hate speech and construct a corpus in advance.

[1] https://github.com/doublems/korean-bad-words

Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH), pages 13–22
December 19, 2021.

In light of this, we study on a hate speech corpus construction scheme that reflects the characteristics of the Korean expressions. Recent works on hate speech corpus have considered social bias as one of the hate components (Waseem and Hovy, 2016; Assimakopoulos et al., 2020) as the hypothesis that bias and hate are closely related (Boeckmann and Turpin-Petrosino, 2002) but hardly labeled it. Though Sanguinetti et al. (2018) performed the most decent work on Italian, we wanted to give more fine-grained analysis on the social bias and stereotype. Referring to the prior works, we create an annotation guideline for Korean entertainment news articles comments where hate speech issues have been prevalent in recent years (McCurry, 2019b,a), and construct annotated corpus through crowd-sourcing. Specifically, we describe bias and toxicity as two main attributes of hate speech and label each of them with three-fold categories.

Throughout the paper, we present the annotation guideline built upon the observation of the comments and corpus construction scheme based on crowd-sourcing. Then, we introduce the corpus characteristics and check whether our procedure and result are adequately accepted within the preceding hate speech studies. The contribution of our study to the field is as follows:

- We observe social bias and toxicity convey hate speech in Korean online news comments and build the hate speech annotation guideline, making the annotated corpus and guideline publicly available.[2]
- We conduct an analysis to find the correspondence of our inductive approach with the preceding sociolinguistic discussions.

2 Data

2.1 Language and Domain

The language of interest in this paper is the Korean online expressions, which are generally variations of written Seoul dialect. We target the news comments that show a lot of informal expressions that are difficult to face in the Sejong corpus (Kim, 2006) or Wikipedia.

For domain, we took a look at the violence of the entertainment news article comments, which triggered the comment overhaul in Korea (Yim, 2020). In entertainment news, not only a target is specified, but also the target is perceived as a representation of a certain group of people, so that many utterances including social bias may appear. Besides, the legal system to regulate hate speech for them has not yet been well settled, and above all, the mental suffering of figures targeted by hate speech is severe.

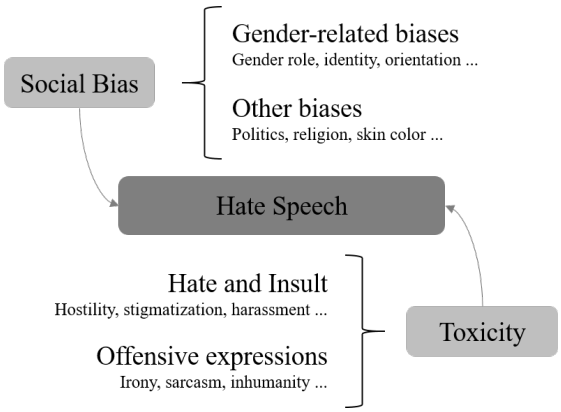

Figure 1: Hate speech attributes in our guideline

2.2 Crawling and Sampling

The articles were crawled for about two years from January 2018 to February 2020, namely the daily top 30 of the online portal news platform, yielding 23,700 articles and 10,403,368 comments.

The following pre-processing was executed to filter the comments.

1. Two articles are sampled daily to avoid bias in a certain period
2. Select the top 20 comments for each article based on the Wilson score (Wilson, 1927) for the downvote
3. Remove duplications
4. Remove (potentially ambiguous) single token comments
5. Remove long sentences over 100 characters

A total of 10,000 comments were collected during this process, along with the head and body of the article, upvotes and downvotes per comment, and a timestamp.

3 Guideline

To create a guideline for large-scale human annotation, three Korean speakers with a computational linguistics background, all familiar with the Korean online materials, read about 1,000 comments

[2]The corpus is disclosed in https://github.com/kocohub/korean-hate-speech with a dataset paper (Moon et al., 2020) and this study focuses more on the annotation guideline, examples, and analysis regarding socioloinguistic discussions.

together, and labeled simultaneously. In our observation, the hate speech that appears in entertainment news comments is judged in terms of "social bias" and "toxicity", as illustrated in Figure 1. Each attribute is identified as follows:

- **Existence of gender-related or other kinds of social bias**[3]
- **Amount of toxicity exposed by hateful, insulting, or offensive expressions**

The main difference between toxicity and bias is that the toxicity is determined based on its intensity (Davidson et al., 2017; Park and Choo, 2017), while for bias, the existence is relatively apparent compared to the former.

3.1 Social Bias

We observe which social bias is disclosed or implicated in each comment. Here, social bias refers to a hasty generalization, stereotyping, or prejudice that an individual/group with a specific social identity will display some characteristics or act in a biased way (Song et al., 2001; Keene, 2011; Blodgett et al., 2020). In the annotation process, the question was labeled in three slots, which are as follows.

3.1.1 Gender bias

Does gender-related bias exist explicitly or implicitly in the text? We considered gender-related biases as primary factors of the social bias, for their prevalence in the Korean online spaces[4] (Kim, 2017b; Lee and Park, 2019). The utterances corresponding to this category include gender role, sexual orientation and identity, and biases for gender-related ideas (Hong et al., 2016; Kim, 2017b). For example, '*Wife must be obedient to the husband's words*' and '*Gays will be vulnerable to disease*' belong to this category. Also, even when gender-related and other biases exist simultaneously in the text, we judged that the case belongs to this category.[5] Brief examples of the subcategories are provided in Korean with English translation, split with '/'. WARNING: This part contains contents that may offend the readers.

- **Prejudice on roles or abilities according to gender**:
  남자는 능력이고 여자는 외모지 *Beautiful women for capable men* / *여자는 집에서 살림하는게 최고, 남자가 부잔가? It's the best for women to stay at home and make beds. The dude's gotta be rich..?*

- **Gender and age**:
  남자가 나이많은 여자를 만날 이유가있나..? 1. 돈줄 2. 성욕해소 말고는.. 글쎄 결혼은 어린여자랑 하는게 맞지 *Why should guys see old women anyway..? 1. for money 2. for spunk release, maybe.. Young ladies are the marriage material*

- **Prejudice against groups with specific gender, sexual orientation, sexual identity, and gender-related ideas**:
  레즈비언들이 좋아할만한.. *Likely to appeal to Lesbians...* / 오래도살았네 게이가 *Talk about longevity for a gay man* / 성전환자가 여자냐? *Transie's a chick?* / 페미들 ㄹㅇ 토나온다 *Feminazis make me sick*

3.1.2 Other biases

Does there exist non-gender-related biases explicitly or implicitly in the text? The utterances of this category incorporate a bias for the factors regarding other social characteristics. In other words, we count the cases that extend an individual's characteristic to a property of a group, so that the speaker's bias towards the group is revealed. These factors include race, ethnicity, nationality, social background, political stance, skin color, religion, disability, age, appearance, richness, occupation, education, and military experience (MacAvaney et al., 2019). Examples of each are as follows.

- **Age**:
  나이에 걸맞게 쳐 놀아라. 이제 마흔이 넘었는데 언제까지나 귀여운 척 할래 ㅉ *Act your freaking age.. Isn't the age of forty a little too old to act all cute?*

[3] In Korean, there are various social deixes that differ according to gender or age, which are commonly used without pejorative purpose. However, there are social movements that try to change or normalize such terms in order to prevent probable offensiveness coming from being referred to as a specific class of gender or age. Since these discussions are ongoing, we could not take into account such sides in making up the annotation guideline. Nonetheless, we believe the change of language may have to be considered as a future improvement of this research.

[4] We first considered that the gender factor had been a central issue regarding hate crime in the previous studies (Jenness, 2003), but some other reasons are to be supplemented in Section 5.4.

[5] This is in concurrence with our decision to separately tag the gender-related bias, and also aims at making the problem a classification.

- **Region of origin**:
  스테이크에 와인한잔하면서 가족끼리 회의했겠지 상도랑 라도는 인생살면서 걸러야한다 *Topic on the table at the family gathering while wining and dining.. Gotta screen them out those Texans and Mississipians*[6]

- **Appearance**:
  돼지같은 것들은...다 이유가 있다 *A pig's a pig for a reason* / 이 외모면 퇴사해도된다 *She doesn't need a job when she looks like that*

- **Occupation**:
  한효주가 뭐가 아쉬워서 사람 많은 클럽에서 침까지 질질 흘리며 콧물까지흘리며 마약을 했을까. 이미지 너무 하락하겠다. *What was wrong with Hyo-Joo Han, for drooling in a crowded club and using drugs? the image will fall too much.* / 존못인데 연기자로서는... *Ugly face, but her acting's okay*

- **Political stance**:
  또 자한당 짓이냐?ㅋ *The Libertarian Party, again. . . ??*[7] / 역시 대깨문답네 *A Trumpist, indeed*[8]

- **When prejudice against a group is involved in judging an individual**:
  무도에서 조금 얻은 인기로 여자 만나고, 그러다 점점 일은 생겼고, 이제는 숨길수 없고 *Get some reputation from the Infinite Challenge*[9]*, then get some girls, then things get messed up, cannot hide* / 경수진 YG소속이네ㅉ YG는 무조건 믿고 거른다ㅉ *Damn, Gyeong Sujin is in YG? B-bye now! I detest all them YGs*[10]

3.1.3 None

The utterances in this category refer to the comments that do not correspond to the two categories above. In the text, prejudice against a group with specific social characteristics is not intervened to judge the group or the individual that belongs to it.

[6]These were originally the profanity terms that denote two main non-capital provinces of Korea, 상도 *Sangdo* and 라도 *Rado*, known to have opposed political stances.

[7]The original expression is 자한당 *Jahandang*, a previous conservative party of Korea.

[8]The original expression is 대깨문 *Daekkaemoon*, initially used for the supporters of the president in Korea, but now used as a swear term to stigmatize and ridicule them.

[9]A variety show of Korea.

[10]YG is a famous entertainment company in Korea, notoriously known for some crimes committed by the members.

3.2 Toxicity

The second attribute is how toxic the comment is, either considering the speaker's intention or the influence on the readers (Wulczyn et al., 2017). The degree of toxicity is a subjective measure, and it is difficult to avoid the annotation being influenced by the annotator's experience and linguistic intuition in this issue. However, to determine the boundary as precise as possible as in Davidson et al. (2017), we categorized the 'intensity' as follows.

3.2.1 Hate or Insult

Is there a strong hate or insult for the target of the article, related figures, or other people? Hate can be seen as an expression in which adversarial and aggressive views towards the aforementioned social characteristics are observed (Hong et al., 2016; Park and Choo, 2017), and insult is a mean expression that can seriously impair the social prestige of a specific figure or group (Kim, 2013).[11]

Here, hate is utilized as a bit different from the one in 'hate speech', which is a slightly more abstract concept. For instance, we can tell that the *toxic comments* are where the *hate speech* is displayed in online spaces, and in deciding some comments as *hate speech*, the attribute of *toxicity* might be taken into account if they contain *hate* or *insult*. Therefore, an expression may be categorized into this type for only including some swear words.

To make this clear, we checked the following properties.

- **Expression that can cause mental pain by severely criticizing or deterring an object**:
  노래실력 제일 거품인 색기 *The worst vocal ever* / 돼지 어쩌구였는데 지워짐 흑흑 ㅜㅜ *What a lardbag* / 노잼에 늙은성 괴들 처노는 방송 *This show is a total bore filled with old nip tucks*

- **Sexual harassment or objectification**:
  겨드랑이도 빨겠다 *I'd even lick her armpit* / 스타킹 벗겨서 발가락빨구시퍼용. *Wanna take off her stockings and suck on those toes* / 보팔 *Pussy chaser*

- **Comments containing hostile feelings toward individuals or groups based on the innate characteristics of individuals/groups**:

[11]Note that the definition here mainly follows the cognates of *hate* and *insult* in Korean articles, while also taking into account the global standard such as Facebook, Youtube, and Twitter.

성 정체성을 잃어가는 병자들이 많은 시대네...병은 고쳐야지 자랑이라고 떠들어 대나? *It is the age of sickos without any proper sexual identity. They need a cure, not a chance to brag about themselves* / 여기 성별에 댓글만봐도 한남 믿거할수있겠다 *I can just look at the gender label and the contents of the replies to pick out the worthless pricks*

- **Expressions intended to negatively stigmatize or define a specific individual/group**:
홍윤화메갈 ?[12] *Hong Yoon Hwa a feminazi?* / 애국보수 산이의 음악행보룰 전폭 지지하는 바입니다. *STAN for San-E and his music career, the true republican nationalist* / 참 대단하다 탑게이 *Applause for the gay lord*[13]

- **Exhibition of the notorious factual events**:
ㅋ 접대와 조작의 아이콘 아이즈원 엑스원 *IZONE X1,*[14] *THE ICON for manipulations and booty calls*

- **Comments showing hostility towards other users**:
언제까지 반일 감정에 불탈래 막상 역사도 그렇게 모르는 개돼지들이 꼭 흥분하더라 *Till when will you be stuck with anti-Japaneseism Morons without any historical knowledge always bark the loudest*

- **Comment showing hostility towards the journalists who wrote the article**:
기레기새끼.의식불명될때까지 쇠파이프로 대가리 깨야됨 *Newshounds*[15] *need to be bashed in the head to the point of unconsciousness* / 인턴기자라는 것이 인턴때부터 제목 좟같이 뽑아서 조회수 늘릴 꼼수를 부리고 있네. 아주 싹이 노랗다 못해 형광색일세. 기레기 꿈나무에 캬얄 퉷 *What a trashy title to come from an intern journo, just dying to get more views. It's too transparent I can even see through it. Here's a finger for the future newshound*

3.2.2 Offensive expressions

Does not reach Insults or hate, but contains aggressive and rude content? The toxicity of these utterances is less than that of hate or insult, but the contents can still make listeners feel offensive. It is expected to be represented by the following properties.

- **Ironic and rhetoric expressions**:
짠내투어 멤버로 랩퍼 도끼를 추천합니다. 근검절약의 아이콘 이시더라고요 *Recommend Doki as a member for Salty Tour.*[16] *Heard he's the man of frugality*

- **Inhumane expressions**:
? 정말 좋아하는 배우 였는다... 가셔서 행복하시길 바라고 갈때가더라도 돈좀 주구가.. 그게 아니면 로또1등좀. *Really liked him/her as an actor/actress.. Well farewell, godspeed, and oh drop some money in my pocket? Or the lottery winning numbers?*

- **Cynical or guessing expression**:
이분 빚투나오는거 아닌지.. 다갚으시고 집자랑 하신거겠죠.. *I'm afraid there will be a #ILoanedHimMoneyToo for the guy. You did pay all your debts before showing off your crib like that, right?* / 송사끝나서 후련한 마음에 동남아 좀 갔다고 뭐 문제라도 있음? *Work complete, you feel nice about it, so took up a trip to Southeast Asia. What's the big deal?*

- **Expressions that can make someone feel bad or demean them**:
무슨 다들 작가들 납셨나봄ㅋㅋㅌㅌㅌㅋ 제발 보지마라 씨부릴거면 ㅋㅋㅋㅋ 각자 취향에 다른거지 난 좋음ㅇㅇ *Wow y'all must be the screenwriters lolololololol just stfu and don't watch it lolololol I like it, everyone's got different tastes. I like it.* / 누군데 얘네? *So who are they?*

- **Comment with no hate, but with abusive language such as swear words**:
한채아를 감히... ㅅㅂ *Fuck, who dare did her?*

3.2.3 None

These refer to comments that do not meet the above toxicity. Even if there is criticism, it is judged as a tolerable opinion in case there is no offensive or rude content. Toxicity is hardly observable in the instances that belong to this.

[12] 메갈 *Megal* is a stigmatized term for the feminists in Korea.

[13] Originally 탑게이 '*Top Gay*', a term that a Korean homosexual entertainer first used to introduce himself.

[14] A Korean Idol group that has been suspected for their agency manipulating the voting system of TV Pro.

[15] The original expression is 기레기 *Giregi* which is a compound of *garbage* and *reporter*.

[16] 도끼 *Doki* is a rapper who is famous for showing off his richness, and 짠내투어 *Salty Tour* is a TV program that aims to travel with as little money as possible.

4 Annotation

The annotation guideline described above was primarily constructed through the analysis of the observed 1,000 comments. However, to help the annotators refer to it in tagging the large-scale corpus, utilization of the crowd-sourcing platform was inevitable. The three factors considered in this process are: 1) *whether the platform has a sufficient number of potential workers*, 2) *whether our guideline can be well taken into account in the annotation process*, and 3) *if the annotation can be performed by the workers that exhibit an expected ethical standard*. Based on these, we adopted *DeepNatural AI*[17] that accommodates a variety of participants, allows pilot study for the selection of the annotators, and supports the system for checking whether the feedback to the annotators is reflected in the resubmission.

Labeling was performed for each attribute through the pilot study and crowd-sourcing, and the decision was made with majority voting. For this, the tagging of three participants were guaranteed for each instance. Additional adjudication was conducted in cases where all three annotators tagged differently or the answers were significantly divided (e.g., when there was no choice in the middle area for the tagging over toxicity).

4.1 Pilot Study

In order for workers of the crowd-sourcing platform to participate in large-scale corpus construction, a pilot study must be performed to ensure the appropriateness of their labeling. We used randomly selected 1,000 comments that were not exploited to make up the guideline, to select the workers who understood our guideline well. The detailed checklist is as follows.

- Is the number of tagging performed more than a certain standard (e.g., 30)?
- Wasn't the omission of tagging too frequent?
- Was the tagging consistently done for challenging instances?
- Was the feedback on the rejected work well reflected in resubmission?
- Isn't the participant exhibiting particular criteria, for gender and other factors, that have significant gaps with the given guideline?

[17] https://app.deepnatural.ai/

(%)	Hate	Offensive	None	Total
Gender	10.15	4.58	0.98	15.71
Other	7.48	8.94	1.74	18.16
None	7.48	19.13	39.08	65.70
Total	25.11	32.66	41.80	100.00

Table 1: The composition of the constructed corpus.

4.2 Crowd-sourcing

We conducted the annotation of left 8,000 comments executed by eight selected participants. Unlike the pilot study that the authors reviewed, rejected, and accepted in a case-by-case manner for selecting the participants, the annotation of the participants was performed on the platform without further restriction.

5 Corpus

The final dataset comes from a total of 10,000 instances, namely those exploited in making up the guideline, the instances reviewed and accepted through pilot studies, and the rest 8,000, crowd-sourced through the annotation of the selected participants. In this process, 659 cases that did not reach the final agreement or was omitted by the participants were dropped.

5.1 Agreement and Performance

The agreement was calculated based on the corpus after adjudication, and based on this, an inter-annotator agreement (IAA) was calculated with Krippendorff's alpha (Krippendorff, 2011). At this time, the agreement on social bias was divided into a binary case that only checks the existence of gender-related bias and a ternary case that separately checks the existence of other biases. The task that detects the existence of a gender-related bias (*gender bias*) shows a relatively high agreement (0.765) compared to the other two cases, while the other two ternary tasks (*social bias* and *toxicity*) showed a moderate but slightly more uncertain label decision (0.492 and 0.496, respectively). The model performance using baseline deep learning architectures is provided in the original dataset paper (Moon et al., 2020), showing the best F1 score of about 0.63 for social bias and 0.58 for toxicity (ternary classification). The agreement and performance show the validity of the proposed corpus.

5.2 Composition

The composition of the whole corpus is as shown in Table 1. Overall, toxic instances occupy a larger volume than those which are not, while the portion of the instances with social bias is comparably smaller than their counterpart. However, it is hesitant to conclude that the toxic comments are more visible in the entertainment news domain than the comments with the bias, since we had collected the comments according to the portion of the downvote. Instead, it may more make sense to interpret that toxicity more influences on judging the comment, compared to the social bias factors which is usually implicated within.

5.3 Analysis

One of the points worth paying attention to is that most of the comments regarding gender-related or other biases are at least offensive or disclose hate/insult in general (toxic among the comments with gender-related bias: 93.76%, toxic among the comments with other bias: 90.42%). On the other hand, it was observed that the toxic comments were not necessarily the ones implicating the social bias.

However, another tendency is displayed in between each social bias type. We were able to discern from the results that the gender-related bias could boost the intensity of the toxicity. That is, we checked that in the comments with higher toxicity (namely hate and insult), the gender-related bias is disclosed about 40% more frequent than other biases (10.15% to 7.48%), while in less toxic (offensive) comments, the tendency is reversed (4.58% to 8.94%).

5.4 Why Gender-related?

The result in Section 5.3 is in concurrence with our premise in the guideline that the gender-related hate speech is more prevalent (Kim, 2017b; Lee and Park, 2019), which is assumed to be in connection with the cultural background (Kim and Lowry, 2005; Koh, 2008; Prieler, 2012). First, as stated in Section 3, considering that the corpus concerns entertainment news article comments which are often in less correlation with other political or social issues, our guideline placed more attention on the gender-related issues, which differs from the previous study that has considered stereotype (Sanguinetti et al., 2018) yet in a binary manner. We directly or indirectly recommended intolerance for gender-related content, for example, by categorizing them separately (social bias) or citing them as a representative example (sexual harassment and sexual insults).

Our approach does not contradict the current data-driven hate speech studies on other languages (Waseem and Hovy, 2016; Fortuna et al., 2019). Our policies were set on purpose since the gender-related factors can influence readers more universally than other contents (such as politics, religion, financial power, etc.), in that the properties are often innate and determine one's identity. That is why other identity factors such as nationality and ethnicity are also crucially investigated in international studies where multiculturalism plays a more significant role (MacAvaney et al., 2019). Those factors are to be further specified and developed in the future guideline.

6 Discussion

Our guideline aims at making the blurry boundary between hate speech and freedom of speech more explicit in order to attack the real-world problem. In this section, we extend how this categorization process can connect with "*which expressions are sociolinguistically defined as hate speech*". We refer to Hong et al. (2016), Kim (2017b), and Park and Choo (2017), where each mainly concerns the definition of hate speech, its target and scope, and the legal issues regarding freedom of speech.

6.1 Definition

Previous studies As stated earlier, in Hong et al. (2016), hate speech is defined as "an expression that discriminates/hates or instigates discrimination/hostility/violence for some social minority individual/group", which follows the National Human Rights Commission of Korea, closer to the European definition (No, 15). Accordingly, the types of hate speech fall into four categories: 1) discriminative bullying, 2) discrimination, 3) disclosed contempt/insult/threat, and 4) hate incitement, where 1-2) are in concurrence *social bias* and 3-4) with *toxicity* defined in this study. Hong et al. (2016) attempts to discern "*this is hate speech*" rather than "*what the hate speech is*", and we expand such factors to the process of corpus construction.

Our approach As described in the guideline, we take the social bias (stereotype, prejudice) and toxicity (hate, insult, contempt, threat) as main attributes, which comes from the typological definition in the deductive approaches (Hong et al.,

2016; Kim, 2017b). The results in Section 5 further suggests that social bias is likely to accompany the toxicity. This is also in concurrence with the discussion that 'discrimination', which is an act of making a distinction based on human identities, is a core factor of hate speech that can represented by bullying, contempt, threat, etc.

6.2 The Borderline of Hate Speech

Previous studies In a slightly distinguished view, Park and Choo (2017) focuses on clarifying the boundary of freedom of expression and hate speech, and aims to establish a principle that can regulate hate speech while minimizing the infringement of freedom. To be concrete, the actual legal cases are examined regarding insults or hate speech, considering which expressions are acknowledged as *violation*. Similarly as in the *definition*, Park and Choo (2017) finds it is challenging to define which expression is clearly illegal. However, Park and Choo (2017) emphasizes the freedom of speech should not be used to justify the attack towards minority or underrepresented groups, and any expressions that infringe on the dignity or personal values of others should be restricted.

Our approach In the previous study, the freedom of speech was taken into account in judging the hate speech (Park and Choo, 2017). In contrast, we made a decision on the toxicity assuming we were the target figure, for instance, how the expression may insult, offense, or mentally harm the addressee. However, since putting on one's shoe is difficult, three annotators' opinion on such perception were aggregated to make up the decision.

One challenging example was the comment that quotes a female celebrity as a sexually attractive figure. This may harass the ordinary female addressee, but since we had limited knowledge of how the target might perceive it, we had to leave it to the annotators' decision. This kind of ethical or social perception is highly dependent on linguistic intuition and experience, and it is also challenging to find well-defined ground truth. We attempted to carefully draw the borderline of biasedness and toxicity in the pilot study, crowd-sourcing, majority voting, and adjudication to guarantee freedom of speech while restricting the social harm (Park and Choo, 2017).

6.3 Minority

Previous study Kim (2017b) refers to the definition of Hong et al. (2016), and describes an *objectiveness* of minority and *mental harm* received by listeners/targets of hate speech. The focus here is whether an utterer denies the identity of the victim via hate speech. It emphasizes the importance of preventing potential victims from facing such violence in open space, and that the society-level response is urgent to this issue.

Our approach In our study, bias and toxicity towards even social dominants (males, the rich, and so on) were regarded as hate speech. Though this may not be harmonized with the previous study (Kim, 2017b), we argue that the mental harm triggered by the hate speech should not be masked out by whether the target is in the majority group or not. Also, the underrepresented group is not always fixed and can change by era.

More importantly, the justification on the bias and toxicity towards the privileged could make unexpected model bias. For instance, what if one indiscriminately insults a public figure just because s/he is rich or educated? How should we handle the inhumane reaction when the victim of a tragedy is male, as in "*Good man is a dead man*"? Again, confining the objectiveness of hate speech to certain minorities may not help detecting real "hateful expressions" from which the victims might suffer. This is also intertwined with the way we draw the borderline, and putting on one's shoe plays an important role here.

7 Conclusion

Throughout this study, we investigated which factors result in hate speech in an inductive way. The hate speech found in Korean online news comments contains either social bias, toxicity, or both. We built a guideline upon the findings and constructed a dataset to train a model that automatically detects them. Furthermore, we refer to the previous discussions on hate speech treated in sociolinguistics and journalism, to see how our approach is related to them and what the distinguished points scrutinized in our approach are.

As a follow-up study, we verified how effective this corpus is as input data for real-problem-solving machine learning model (Moon et al., 2020), and will check whether its detection performance is affected by (maybe biased) pre-trained language models. Also, we will investigate how the construction scheme of our corpus can be leveraged in other domain such as depressive online text detection

(Hämäläinen et al., 2021). Besides, we expect that by the release of this corpus, Korean hate speech research is to be diversified and that real online space might be detoxified.

Acknowledgments

Authors thank Jumbum Lee for creating the dataset together and appreciate Hyunjoong Kim for aiding the crowd-sourcing process. Also the authors are grateful for encouraging comments from two anonymous reviewers.

References

Stavros Assimakopoulos, Rebecca Vella Muskat, Lonneke van der Plas, and Albert Gatt. 2020. Annotating for hate speech: The MaNeCo corpus and some input from critical discourse analysis. pages 5088–5097.

Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. 2019. SemEval-2019 Task 5: Multilingual detection of hate speech against immigrants and women in Twitter. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 54–63.

Su Lin Blodgett, Solon Barocas, Hal Daumé III, and Hanna Wallach. 2020. Language (technology) is power: A critical survey of "bias" in NLP. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5454–5476, Online. Association for Computational Linguistics.

Robert J Boeckmann and Carolyn Turpin-Petrosino. 2002. Understanding the harm of hate crime. *Journal of Social Issues*, 58(2):207–225.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *Eleventh International AAAI Conference on Web and Social Media*.

Mai ElSherief, Vivek Kulkarni, Dana Nguyen, William Yang Wang, and Elizabeth Belding. 2018. Hate lingo: A target-based linguistic analysis of hate speech in social media. In *Twelfth International AAAI Conference on Web and Social Media*.

Facebook. Facebook's policy on hate speech. https://www.facebook.com/communitystandards/hate_speech. Accessed: 2020-10-06.

Paula Fortuna, João Rocha da Silva, Leo Wanner, Sérgio Nunes, et al. 2019. A hierarchically-labeled Portuguese hate speech dataset. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 94–104.

Mika Hämäläinen, Pattama Patpong, Khalid Alnajjar, Niko Partanen, and Jack Rueter. 2021. Detecting depression in Thai blog posts: a dataset and a baseline. In *Proceedings of the Seventh Workshop on Noisy User-generated Text (W-NUT 2021)*, pages 20–25, Online. Association for Computational Linguistics.

Sung Soo Hong et al. 2016. *Study on the State and Regulation of Hate Speech*. National Human Rights Commission of Korea.

Valerie Jenness. 2003. Engendering hate crime policy: Gender, the "dilemma of difference," and the creation of legal subjects. *Journal of Hate Studies*, 2(1):73–92.

Seungche Kang. 2018. *A study on constructing dictionary for Korean hate speech classification: Focusing on online news comments*. Korea Advanced Institute of Science and Technology.

Sabrina Keene. 2011. Social bias: Prejudice, stereotyping, and discrimination. *The Journal of Law Enforcement*, 1(3):2–4.

Bo-Myung Kim. 2018. Late modern misogyny and feminist politics: The case of Ilbe, Megalia, and Womad. *Journal of Korean Women's Studies*, 34(1):1–31.

Doo Sang Kim. 2013. A study on the regulations of defamation and insult on cyberspace. *The Journal of Legal Studies*, 21(1):175–196.

Hansaem Kim. 2006. Korean national corpus in the 21st century Sejong project. In *Proceedings of the 13th NIJL International Symposium*, pages 49–54. National Institute for Japanese Language Tokyo.

Jinsook Kim. 2017a. # iamafeminist as the "mother tag": Feminist identification and activism against misogyny on Twitter in South Korea. *Feminist Media Studies*, 17(5):804–820.

Kwangok Kim and Dennis T Lowry. 2005. Television commercials as a lagging social indicator: Gender role stereotypes in Korean television advertising. *Sex Roles*, 53(11-12):901–910.

Sooah Kim. 2017b. Expression of hate and discrimination in the Korean language from a social viewpoint: Problem statement and improvement measures for hate and discrimination against social minorities. *New Korean Language-life*, 27(3):49–63.

Eunkang Koh. 2008. Gender issues and Confucian scriptures: Is Confucianism incompatible with gender equality in South Korea? *Bulletin of the School of Oriental and African Studies*, 71(2):345–362.

Klaus Krippendorff. 2011. Computing Krippendorff's Alpha-reliability. *Computing*, 1:25–2011.

Young-Joo Lee and Ji-Young Park. 2019. Emerging gender issues in Korean online media: A temporal semantic network analysis approach. *Journal of Contemporary Eastern Asia*, 18(2):118–141.

Sean MacAvaney, Hao-Ren Yao, Eugene Yang, Katina Russell, Nazli Goharian, and Ophir Frieder. 2019. Hate speech detection: Challenges and solutions. *PloS one*, 14(8):e0221152.

Justin McCurry. 2019a. K-pop singer Goo Hara found dead aged 28. *The Guardian*.

Justin McCurry. 2019b. K-pop under scrutiny over 'toxic fandom' after death of Sulli. *The Guardian*.

Jihyung Moon, Won Ik Cho, and Junbum Lee. 2020. BEEP! Korean corpus of online news comments for toxic speech detection. In *Proceedings of the Eighth International Workshop on Natural Language Processing for Social Media*, pages 25–31, Online. Association for Computational Linguistics.

ECRI General Policy Recommendation No. 15. on combating hate speech, 8 December 2015.

Da-Sol Park and Jeong-Won Cha. 2018. Semi-supervised learning for detecting of abusive sentence on Twitter using deep neural network with fuzzy category representation. *Journal of KIISE*, 45(11):1185–1192.

Mi-suk Park and Ji-hyun Choo. 2017. *The State of Hate Speech and The Response Measures*. Korean Institute of Criminology.

Michael Prieler. 2012. Gender representation in a Confucian society: South Korean television advertisements. *Asian Women*, 28(2):1–26.

Manuela Sanguinetti, Fabio Poletto, Cristina Bosco, Viviana Patti, and Marco Stranisci. 2018. An Italian Twitter corpus of hate speech against immigrants. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Kwan Jae Song, Jae Chang Lee, and Young Oh Hong. 2001. Prejudices and discrimination toward social stigmatized groups. *Korean Journal of Psychological and Social Issues*, 7(1):119–136.

Twitter. Twitter's policy on hate speech. https://help.twitter.com/en/rules-and-policies/hateful-conduct-policy. Accessed: 2020-10-06.

Zeerak Waseem and Dirk Hovy. 2016. Hateful symbols or hateful people? predictive features for hate speech detection on Twitter. In *Proceedings of the NAACL Student Research Workshop*, pages 88–93.

Edwin B Wilson. 1927. Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Association*, 22(158):209–212.

Ellery Wulczyn, Nithum Thain, and Lucas Dixon. 2017. Ex machina: Personal attacks seen at scale. In *Proceedings of the 26th International Conference on World Wide Web*, pages 1391–1399.

Hyunsu Yim. 2020. Why Naver is finally shutting down comments on celebrity news. *The Korea Herald*.

Youtube. Youtube's policy on hate speech. https://support.google.com/youtube/answer/2801939?hl=en. Accessed: 2020-10-06.

MacBERTh: Development and Evaluation of a Historically Pre-trained Language Model for English (1450-1950)

Enrique Manjavacas
Leiden University
Leiden, The Netherlands
enrique.manjavacas@gmail.com

Lauren Fonteyn
Leiden University
Leiden, The Netherlands
l.fonteyn@hum.leidenuniv.nl

Abstract

The new pre-train-then-fine-tune paradigm in Natural Language Processing (NLP) has made important performance gains accessible to a wider audience. Once pre-trained, deploying a large language model presents comparatively small infrastructure requirements, and offers robust performance in many NLP tasks. The Digital Humanities (DH) community has been an early adapter of this paradigm. Yet, a large part of this community is concerned with the application of NLP algorithms to historical texts, for which large models pre-trained on contemporary text may not provide optimal results. In the present paper, we present "MacBERTh"—a transformer-based language model pre-trained on historical English—and exhaustively assess its benefits on a large set of relevant downstream tasks. Our experiments highlight that, despite some differences across target time periods, pre-training on historical language from scratch outperforms models pre-trained on present-day language and later adapted to historical language.[1]

1 Introduction & Related Work

Social scientists and Humanities scholars have long been interested in describing cultural systems and understanding the way in which these change across time. Traditionally, such shifts were documented with 'manual' interpretative methods, but more recently researchers in DH have begun applying Machine Learning techniques to support their interpretation.

In the case of researchers working with historical text, current work has been occupied with developing and evaluating NLP algorithms with the goal of modeling the way in which concepts, categories and discourses (e.g. of class, gender) change over time along with their linguistic representations.

In this context, applications include data-driven approaches to conceptual change (Fitzmaurice et al., 2017; Sommerauer and Fokkens, 2019; Marjanen et al., 2019; Martinez-Ortiz et al., 2019), historical word sense disambiguation (Bamman and Crane, 2011; Fonteyn, 2020; Beelen et al., 2021), Named-Entity Recognition in historical text (Labusch et al., 2019; Konle and Jannidis, 2020; Schweter and Baiter, 2019; Schweter and März, 2020; Ehrmann et al., 2020; Boros et al., 2020) or unsupervised semantic change (Schlechtweg et al., 2020; Giulianelli et al., 2020).

In view of the growing weight of the new NLP paradigm of "pre-train-and-fine-tune"—which leverages large language models in order to produce strong feature extractors (Peters et al., 2018; Radford et al., 2018; Devlin et al., 2019)—, the question arises as to whether similar performance boosts can be gained for current NLP-oriented research dealing with historical text.

Due to the heavy domain-shift along grammatical, semantic and orthographic language layers, models pre-trained on contemporary data are less helpful when applied on historical data. Previous work has experimented with adapting contemporary models to historical data on a per-task basis (Han and Eisenstein, 2019), although it is unclear whether this approach can yield a general purpose historical model. An alternative approach would be to adapt historical text to the modern standards using a historical text normalization system (Bollmann, 2019). While this could indeed help tackling orthographic shifts, grammatical and semantic shifts would be left un-adapted. Moreover, error percolation from the automatic normalization system would still be an issue.

Arguably, the main advantage of pre-training

[1] Evaluation code is available through the project's repository: https://www.github.com/emanjavacas/macberth-eval. "MacBERTh" itself is available as emanjavacas/MacBERTh from the transformers repository (Wolf et al., 2019).

Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH), pages 23–36
December 19, 2021.

this type of models is their ability to exploit very large datasets. In the case of historical linguistic resources, the known scarcity of digitized text—even for commonly high-resource languages like English—turns such enterprise problematic. However, ongoing efforts towards making historically relevant book collections digitally accessible (Langley and Bloomberg, 2007; Labs, 2014; Mueller et al., 2016) have kick-started experimentation in this respect. For example, Bamman and Burns (2020) pre-trained a model on Latin text spanning several centuries. Schweter and Baiter (2019) and Konle and Jannidis (2020) have employed contextualized character-level models, (Schweter, 2020) has released historical German and French models trained on historical newspaper data, and Beelen et al. (2021) and Hosseini et al. (2021a) trained and released models on an English corpus spanning the 18th to 20th centuries.

Several questions appear in this context. For example, it remains unclear whether all target periods can benefit equally from a "historically" pre-trained model or whether the performance benefits of these models vary across periods depending on the available amount and type of documents. Moreover, it is unclear what the advantages are of the two current alternative pre-training approaches. In some cases, pre-existing models pre-trained on contemporary datasets are first "historically fine-tuned" before being applied on downstream tasks. This approach—motivated by the promise to leverage a larger out-of-domain contemporary dataset—has been shown to outperform their non-adapted counterparts (although, see German BERT vs. Europeana BERT in Schweter and März, 2020), but it remains unclear how these fine-tuned models compare to models pre-trained "historically" from scratch, and, more generally, whether the presence of modern data in the training process diminishes model performance (as suggested by Boros et al., 2020).

Contributions In this paper, we introduce a model pre-trained on a large span of historical English (1450-1900), and show its advantages with respect to present-day models, as well as models adapted from present-day to historical English on an exhaustive set of ad-hoc downstream tasks. Moreover, we show how model performance strongly depends on the time period of the target application.

2 Experimental Setup

We rely on the large language model known as "BERT"—a stack of transformer layers with a self-attention mechanism (Vaswani et al., 2017), optimizing a Masked Language Model (MLM) objective (Devlin et al., 2019). Despite the existence of several MLM alternatives, BERT remains a good choice, considering that (i) it is well-established and most thoroughly studied, and (ii) on-going evaluation of alternative choices—mostly focus on Natural Language Understanding (NLU) tasks—has not yielded a clearly superior architecture.

We rely on the seminal implementation,[2] with the hyper-parameterization corresponding to the "BERT-base Uncased" architecture.[3] Pre-training is done with default parameters, except for the maximum sequence length (set to 128 subtokens) for 1,000,000 training steps.

2.1 Pre-training Dataset

The model is pre-trained on a corpus of a total size of ca. 3.9B (tokenized) words (time span: 1450-1950) using the following corpora: the Early English Books Online (EEBO) corpus, the Corpus of Late Modern English Texts (CLMET3.1), the Evans Early American Imprints Collection (EVANS), Eighteenth Century Collections Online (ECCO), the Corpus of Historical American English (COHA), and the Hansard corpus (Hansard). The resulting corpus is a varied collection in terms of text types, including literary works, religious and legal text as well as news reports and transcriptions of British parliamentary debates. The summary word count statistics are shown in Figure 1.

In terms of preprocessing, the corpus was first cleaned up in order to remove foreign text,[4] and split into sentences using the NLTK built-in sentence tokenizer (Bird, 2006).

2.2 Benchmarking

To cast light upon the advantages of large MLMs for diachronic tasks, we designed a number of benchmarking tasks, in which the contextualized

[2] Available on the following URL: https://github.com/google-research/bert.

[3] See Section 2.2.2 or the original paper for a description of these parameters.

[4] We used an ensemble of the Google's Compact Language Identifier (v3) and the FastText Language Identification system (Grave, 2017), operating over chunks of 500 characters, which were flagged as foreign whenever both systems indicated a language other than English as the highest probability language.

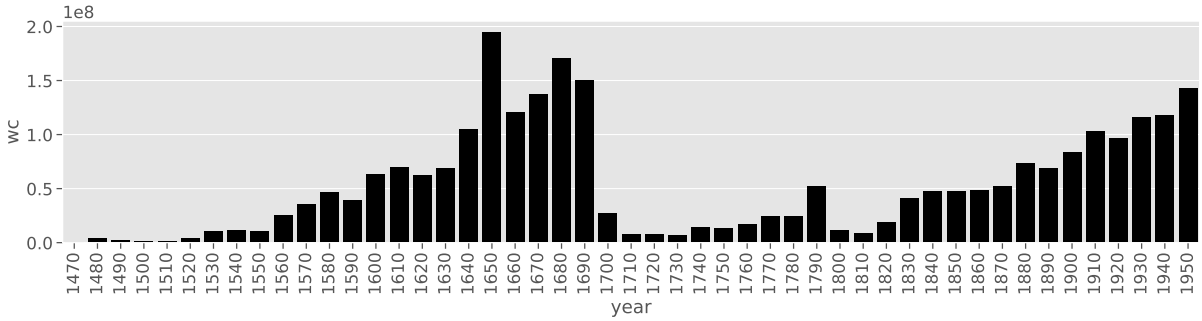

Figure 1: Aggregated word count statistics per decade for the pre-training corpora used in the present study.

representations of candidate models must encode historical information in order to achieve strong performance.

2.2.1 Benchmarking Datasets

In line with previous work (Hu et al., 2019; Heo et al., 2020; Beelen et al., 2021), we rely on data from the Oxford English Dictionary (OED Simpson and Weiner, 1989)—an authoritative resource for historical and contemporary lexical semantics in the English lexicon—for the benchmarking tasks. For each lemma, the OED defines a hierarchy of word senses, including quotations exemplifying each sense over the entire historical span of that sense. For the present experiments, we sampled 3,000 words from the vocabulary of the corpus described in Section 2.1, in proportion to their smoothed relative frequencies. Each word was retrieved and matched to existing lemmata in the OED reservoir. Upon successful retrieval, the senses and quotations of the corresponding lemma were stored. The resulting dataset comprises 2,700 lemmas, 35,110 senses and 246,048 quotations, which we utilize in varied ways for benchmarking.

We also include part-of-speech tagging, using the Penn-Helsinki Parsed Corpus of Early Modern English (PPCEME) (Kroch et al., 2004)—a manually annotated corpus of Early Modern English (time span: 1450-1700), comprising about 1.7m words over 448 documents.[5]

2.2.2 Candidate Models

In order to quantify the relative advantage of pre-training MLMs on historical corpora, we compare different instantiations of BERT. First, we benchmark against models with comparable architectures trained on **present-day English data** only. We use `BERT`, which corresponds to "BERT-Base Uncased" in the original repository, and is trained on ca. 3.3B tokens—i.e. the BookCorpus (Zhu et al., 2015) and the English Wikipedia—using a WordPiece (Schuster and Nakajima, 2012) vocabulary of 30,000; and `MultiBERT`, which corresponds to "BERT-Base Multilingual Cased" from the original repository, and is trained on the union of the top 100 languages in terms of the size of the respective Wikipedia sites, using a shared WordPiece vocabulary of 110,000.

Secondly, we compare with a variant of `BERT`—i.e. "BERT-Base Uncased"—that was fine-tuned at the Alan Turing Institute on 5.1B tokens of **historical English** (time span: 1760-1900 Hosseini et al., 2021a),[6] which we label `TuringBERT`.

Contemporary `BERT` and `MultiBERT` differ mainly in training material and vocabulary. `MultiBERT` should have an advantage when applied to historical English data, as it is trained on a much larger and varied dataset and with a more flexible vocabulary. The main difference between `TuringBERT` and `MacBERTh` is their span and size, with `TuringBERT` covering a smaller time window but a larger training dataset. Furthermore, as `MacBERTh` was pre-trained from scratch, its vocabulary is better adjusted to historical English.

3 Results

We now describe the benchmarking tasks and the results of the competing models in detail.

3.1 Part-of-speech Tagging

The first task tackles part-of-speech tagging of historical documents. Historical text is known to be challenging for automatic processing due

[5]We replicate the training-test splits from (Han and Eisenstein, 2019), thus keeping 115 files for testing and reserving from the 333 remaining 17 randomly sampled files (ca. 5%) for development purposes.

[6]The model is available through the accompanying online repository (Hosseini et al., 2021b).

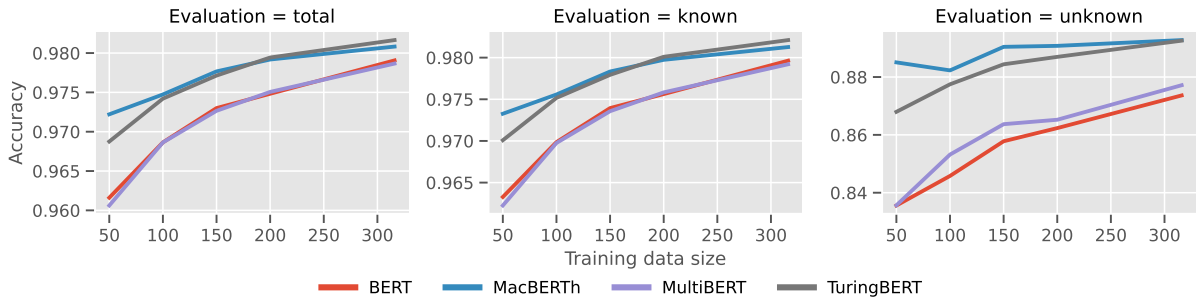

Figure 2: Accuracy on the part-of-speech tagging task (on the y-axis) considering different training size regimes (in number of documents in the x-axis) and different evaluation criteria: total (all tokens), known (only tokens observed in the training set) and unknown (only tokens not present in the training set).

to (relatively more) complex inflection systems and lacking orthographic standards (Manjavacas et al., 2019). As recently shown by Han and Eisenstein (2019), domain-specific pre-training yields improvements for historical pos-tagging, even if no labeled data is available for the target domain. We quantify the potential of pre-training on historical data for pos-tagging of historical texts by computing accuracy on held-out data after fine-tuning the pre-trained MLMs on **target-domain** data.[7]

We expect pre-training on target domain data to improve tagging accuracy of documents, especially if these stem from the same period. Moreover, in line with (Han and Eisenstein, 2019) we particularly expect improvements for tokens in the held-out data that were not encountered during training. Finally, we also test the relative sample efficiency of the competing MLMs by fine-tuning these on incrementally smaller samples of training data. To test our hypotheses, we compute (micro-)accuracy of all, known and unknown tokens, using random sub-samples (50, 100, 150, 200 and all files).[8]

Pre-training on contemporary material (`BERT`, `MultiBERT`) results in less accurate models for all evaluation conditions (see Figure 2). The historical models have an advantage, especially on unknown tokens. The model pre-trained on the larger temporal span (i.e. `MacBERTh`) has an advantage in the smaller training data regime.

Moreover, when factoring in the date of the held-out document, we find that the relative improvement of `MacBERTh` is larger for earlier dates, and seems to increase for later dates, as shown in Figure 3 for the accuracy of unknown tokens in the small-size training regime. This may be explained by the combined effect of the sample efficiency of the different models and the training data size of the different periods (shown in Figure 3 by the shaded grey area), which is seemingly correlated with the advantage of `MacBERTh`.[9]

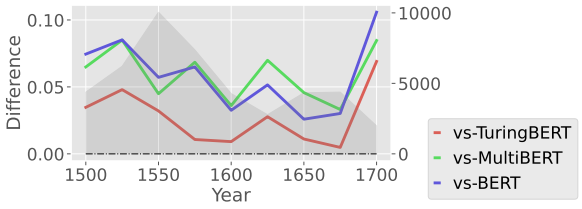

Figure 3: Difference in part-of-speech tagging accuracy of unknown tokens of `MacBERTh` with respect to the alternative models in the small-size training regime (50 documents). The shaded area shows the total number of tokens per period on which the evaluation is based.

3.2 Word Sense Disambiguation

Word Sense Disambiguation (WSD) by modeling lexical semantics in context has received ample attention from the DH community—see the different applications surveyed by Tahmasebi et al. (2018, Section 7.1 and 7.2)—, and is arguably one of the most promising venues for deploying MLMs.

We first approach WSD as a binary classification task in which pre-trained models are fine-tuned in order to predict whether a pair of quotations exemplifying senses of a given lemma correspond to the same sense or not (Section 3.2.1). Second, we evaluate the quality of sense embeddings derived from the MLMs without explicit fine-tuning in a sense classification task (Section 3.2.2).

[7] We fine-tune all MLMs on the PPCEME for 3 epochs with a batch size of 8 on a single GPU.

[8] The random sub-samples of training data were kept constant for all models.

[9] For the sake of completeness, the full results are shown in Figure 10 in the Appendix.

3.2.1 Word-in-Context

The word-in-context task for evaluating context-sensitive word representations was introduced by Pilehvar and Camacho-Collados (2019), following earlier efforts on evaluating context-dependent word similarity (Huang et al., 2012). Recently, Beelen et al. (2021) have also focused on this task, referring to it as "targeted sense disambiguation".

We utilize the OED quotations dataset from Section 2.2.1 in the following manner. First, we drop quotations with less than 5 words, in order to ensure that there is enough context for disambiguating. Second, we drop lemmata with less than 100 quotations left as well as lemmata that do not correspond to nouns, adjectives and verbs (based on the OED's lemma categorization). From the resulting dataset of 408 lemmata we reserved 10% (= 41), which are used for testing the generalization capabilities of the models. For a given input quotation, we generate a positive example by sampling a paired quotation belonging to the same sense and a negative example by sampling a quotation from a different sense of the same lemma.

In order to fine-tune the models, we replicate the settings in Devlin et al. (2019, Section 4.1), using the last hidden activation corresponding to the `[CLS]` token, adding a linear projection layer in order to compute the logits of the positive and negative class, and optimizing a cross entropy loss. In order to let the model focus on the word that corresponds to the underlying lemma, we add `[TGT]` tokens around the focus word in both members of the input pair.[10,11]

Table 1 shows the results of development (= 25% of the training data) and held-out data. For each block of results, we further distinguish whether the instantiation of the lemma corresponds to the same part-of-speech tag, and, in case of correspondence, we report results per part-of-speech tag—i.e. noun (N), adjective (Adj) or verb (V).

Overall, `MacBERTh` obtained the best results across conditions, except for held-out adjectives where contemporary models had an advantage. Performance on development data is generally very high, surpassing 90% accuracy across conditions. However, on held-out lemmata, no model surpasses 70% accuracy (with adjectives being easier to classify than nouns and verbs for all models).

Model	Total	≠ POS	= POS		
			N	Adj	V
Development					
`BERT`	89.9	92.3	90.1	92.6	86.7
`MultiBERT`	92.0	94.8	91.6	95.6	88.7
`TuringBERT`	91.0	94.1	90.5	94.3	87.6
`MacBERTh`	**94.5**	**96.1**	**94.1**	**96.8**	**92.6**
Held-out					
`BERT`	59.5	56.8	60.5	65.8	58.4
`MultiBERT`	62.1	63.3	64.0	**66.1**	57.1
`TuringBERT`	58.6	58.5	59.8	60.7	56.4
`MacBERTh`	**63.0**	**63.8**	**65.3**	61.8	**59.3**

Table 1: Results of the word-in-context task for development and held-out lemmata across different conditions.

Pairs with diverging part-of-speech tags resulted in higher accuracy, which can be explained by class imbalance: pairs with diverging tags tend to belong to different senses. Interestingly, the drop in performance for the positive class with respect to the negative class was much smaller for `MacBERTh` (6.9 points) than for the other models (9.8 for `BERT`, 9.9 for `MultiBERT` and for 12 points for `TuringBERT`), thus suggesting a stronger generalization ability of `MacBERTh` over competitors.

Finally, we observe that the afore-mentioned advantage is not evenly distributed over the periods from which the input quotations stem. Instead, the advantage of `MacBERTh` was generally larger for quotations originating before the 1700s.[12]

3.2.2 Parameter-free WSD

In the full-fledged WSD setting, an input quotation must be tagged with the sense that it is exemplifying. A further difference with the word-in-context task is that we do not use any additional fine-tuning in order when approaching the task. Instead, we follow the approach outlined in Peters et al. (2018, Section 5.3). The distributed representations of senses are first computed, and then a sense is predicted for an unseen input quotation based on its proximity to the different sense representations (using the nearest sense representation neighbor in terms of cosine similarity). We restrict ourselves to a centroid approach to building sense representations, in which the contextualized vectors of the

[10] An example input pair can be seen in Table 3 in the Appendix.

[11] We use the "sbert" library (Reimers and Gurevych, 2019) to fine-tune the models, training for 5 epochs with batch size of 16 on a single GPU.

[12] For completeness, a full visualization of the difference in accuracy over time bins can be seen in Figure 11 (Appendix).

Model	Total	Word Type	
		Content	Function
BERT	36.0	36.1	34.8
MacBERTh	**42.3**	**42.0**	**50.4**
MultiBERT	32.3	32.1	38.6
TuringBERT	34.8	34.6	43.2
Majority	13.6	13.7	9.1
Random	9.2	9.3	6.1

Table 2: Results of the WSD comparison in terms of classification accuracy by word type.

target tokens exemplifying a particular word sense are averaged.

In order to build a dataset, we utilize the OED quotations from Section 2.2.1. We first drop lemmata with less than 50 quotations. Second, we discard single-sense lemmata as well as senses (of a given lemma) with less than 2 quotations (as we cannot produce classifications in those cases). On the basis of the remaining senses, we generate a stratified training and test set split with 50% of the quotations in each set. In order to classify the sense of an input sentence, we only need to compare it against the sense representations of the same lemma. For this purpose, we rely on the original OED's lemmata, thus assuming gold lemmata.

The results, split by word type, are shown in Table 2.[13] MacBERTh outperforms the competitors across all conditions. Overall, models performed better on function words than on content words, even though the latter seems to be an easier task as per the baseline. Interestingly, TuringBERT is outperformed by BERT, despite the former being fine-tuned on historical material.

Figure 4 factors in time on the x-axis, showing that the effect of time on accuracy is constant across models for content words. In the case of function words, the historical pre-training of MacBERTh seems to be of benefit in the earlier periods.

3.3 Fill-in-the-blank

The ad-hoc fill-in-the-blank task indirectly tackles NLU. For a given OED input, we mask the target token (i.e. the token corresponding to the word of which a sense is being exemplified) and interpret the plausibility assigned by the model to the target token as a proxy of the model's strength to capture

[13] We consider function words those tagged with "pron", "prep", "conj" or "int" following OED's classification.

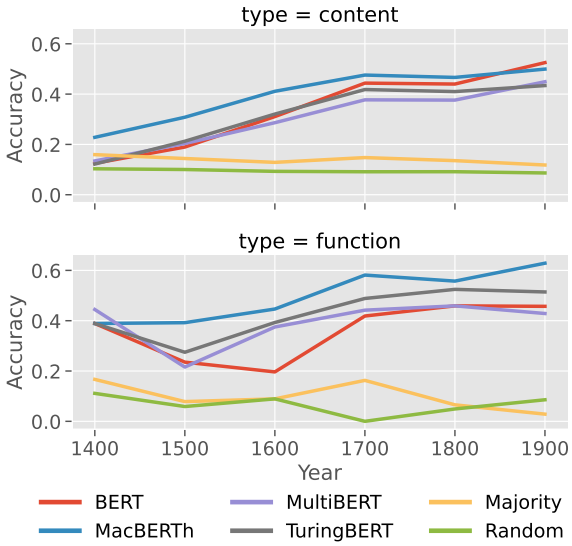

Figure 4: Results of the parameter-free WSD classification experiment by word type and year, including results for Majority and Random baselines.

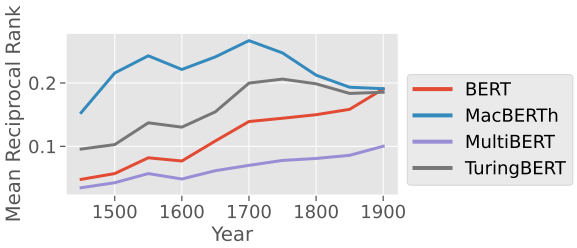

Figure 5: Results of the fill-in-the-blank task over time in terms of Mean Reciprocal Rank.

meaning.

Using the OED quotations from Section 2.2.1, we first select quotations for which the target token is part of the vocabulary of all compared models, which ensures that the comparison is fair. Moreover, since models differ in vocabulary sizes, we evaluate using the rank of the target token based on the logits (instead of directly using the full output distribution of logits). We use the Mean Reciprocal Rank as evaluation metric, averaging over quotations—shown in Figure 5.

The MacBERTh model tops across all periods. The difference with respect to the other models is larger in the earlier periods, highlighting a stronger ability to capture the semantics of earlier examples.

3.4 Sentence Periodization

The last task concerns periodizing quotations from the OED. OED quotations constitute a particularly well-suited test bed, as they have been selected by the OED editors in order to exemplify particular

word usages within specific diachronic frames. A periodizing model, thus, may exploit not only formal aspects of how English has changed (such as spelling, morphology, word order or grammar) but also changes in lexical semantics.

To tackle this task, we first fine-tune the different MLMs in a binary classification task with the goal of predicting whether the first sentence stems from a later period than the second. We deploy the same architecture as the one described in Section 3.2.1 but drop the signalization of the target token.

In order to periodize an input sentence, we use a subset of sentences for which the dates are known (we refer to this subset as the "background corpus"). This subset is both representative of the entire time range for which predictions need to be produced (i.e. sampled uniformly over equally sized spans in the OED), and kept apart during training. Then, for a given input sentence, we obtain a distribution of scores (i.e. probability) over years by comparing the input sentence against sentences from this background corpus.

For each background sentence, the model yields a probability that the input sentence stems from a later period. We first sort these probabilities by the years corresponding to the sentences in the background corpus. We then compute the cumulative distribution (which draws a strictly increasing curve). The prediction then corresponds to the point of maximum curvature or **knee point** within this curve, which we compute using the Kneedle method described by Satopaa et al. (2011). This method identifies the highest point in the curve after (i) smoothing out edges using a polynomial fit of the input data points and (ii) rotating the curve so that both the start and end point lie on the same horizontal line. An example prediction using this method is shown in Figure 6.

We use the OED data from Section 2.2.1, removing quotations with less than 5 words. From the remaining set, we reserve 5% for development and 5% for testing. The remaining 90%, is randomly split into 75% for training and 25% for the background corpus (which is produced by binning the range from 1450 to 1900 into decades and sampling 20 quotations per decade, giving a background corpus of 1,000 quotations in total). Finally, the training, test and development splits are turned into datasets by generating random pairs, ensuring that quotations in the input pairs do not belong to the same lemma. We restrict ourselves to 100,000 training and 5,000 development and test input pairs.[14]

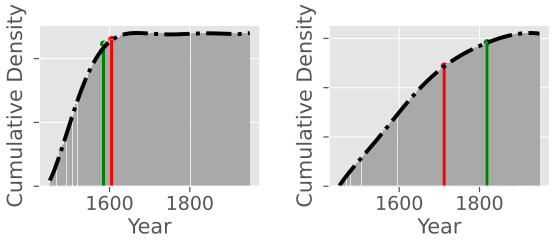

Figure 6: Visualization of a sentence periodization prediction using the knee method. The dashed line shows the cumulative distribution of prediction scores of a given input sentence with respect to the background corpus (x-axis). The grey line corresponds to the smoothing derived from a 7^{th} degree polynomial fit. Finally, the green and red lines highlight the true and predicted year, respectively. Left and right plots show examples of an accurate and inaccurate prediction.

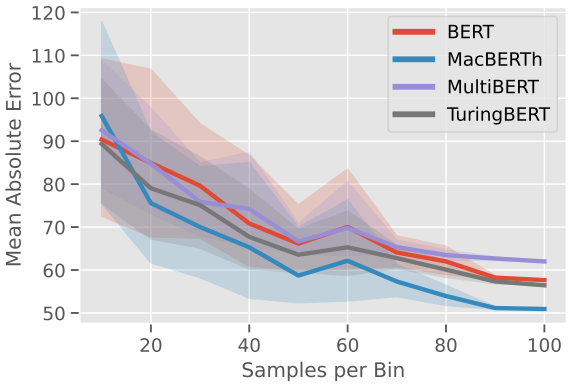

Figure 7: Visualization of the Mean Absolute Error achieved by the different models (lower is better) with respect to the number of samples in the background corpus.

Figure 7 shows the results in terms of Mean Absolute Error (MAE). As the size of the background corpus is a source of variation, we re-run the experiment varying the number of background instances within each 50 year bin (shown on the x-axis), until reaching the full size of 1,000 background quotations (i.e. 100 instances for each of the 10 bins). All models converge to their optimum performance when using the full background corpus. Figure 7 also shows that `MacBERTh` has the smallest error, being wrong on average by 50 years. `TuringBERT` is on par with `BERT` and outperforms `MultiBERT`.

Figure 9 factors in the time dimension, aggre-

[14] We fine-tune the models following the same setting as in Section 3.2.1.

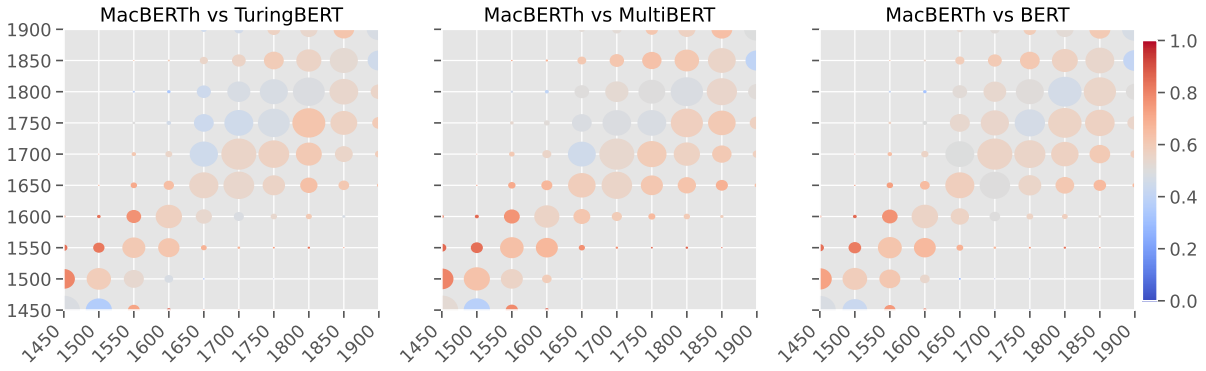

Figure 8: Visualization of the difference in performance of the competing models with respect to MacBERTh. Each circle corresponds to predictions in which the models being compared diverge. The color indicates the proportion of diverging predictions in which MacBERTh is right (thus, red and blue indicates whether MacBERTh is predominantly right). The size of each circle is proportional to the total number of diverging predictions. The y-axis and x-axis indicate respectively the period of the left and right input sentences.

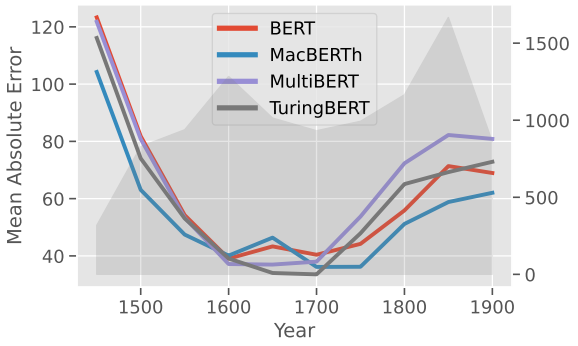

Figure 9: Mean Absolute Error over bins of 50 years. The shaded area shows the total number of pairs per period on which the evaluation is based.

gating MAE over bins of 50 years. MacBERTh achieves the best results for input sentences dated before 1650, as well for those dated after 1750. Further, Figure 8 shows the comparison of MacBERTh vs. the rest taking into account the source year bin of the left (y-axis) and right (x-axis) input examples. All models agree on the periodization of sentences that are separated in time by a larger span—i.e. off-diagonal bins are sparsely populated, indicating a small number of divergent predictions—especially when sentences prior to 1650 enter the comparison. MacBERTh's improvement over the other models seems to be concentrated in the period before 1650 (as suggested by the more intense color in the corresponding part of the plot).

4 Discussion

The exhaustive set of benchmarking experiments allows us to assess the impact of pre-training MLM architectures on historical data for diachronic tasks. As expected, historical pre-training helps to improve performance on diachronically relevant tasks. Accordingly, both TuringBERT and MacBERTh generally outperformed the models pre-trained on contemporary data only—with the exception of both WSD tasks, where MultiBERT (in word-in-context) and BERT (in the parameter-free WSD) outperformed TuringBERT. The historically pre-trained MacBERTh outperformed all competing models across tasks on partitions of the test data stemming from earlier periods.

Furthermore, based on the fact that MacBERTh displays an advantage across tasks and conditions, we can conclude that enlarging the historical span and coverage of pre-training data is advantageous.[15] Importantly, the period to which only MacBERTh had access during pre-training coincides roughly with the beginning of Late Modern English (around the 1700s) and the consolidation of the modern standard. Therefore, if variety in the pre-training data sources results in more powerful feature extractors, the impact for diachronic downstream tasks of the pre-training data stemming from before the 1700s is even so larger.

Interestingly, in many tasks, the advantage shown by MacBERTh was not restricted to the earlier periods. For example, in part-of-speech tagging, the increase in accuracy appeared to be correlated with the total amount of data available for training and testing. This can be interpreted as

[15] Note that this result was obtained even when the total token counts of the pre-training dataset of MacBERTh was smaller with respect to the other models.

a sign of better sample efficiency that pre-training on varied datasets confers the model.

Finally, a question that arises from the present experiments concerns those aspects of the experimental setting that may be responsible for the observed disadvantage of `TuringBERT` on diachronic tasks—even on time spans to which `TuringBERT` had access during pre-training. The fact that the tokenizer is restricted to the specific domain of contemporary English may force the model to aggregate over odd subword tokenizations in order to extract word-level feature representations, putting it in a weakened position. Adapting a model originally pre-trained on contemporary English may also import too strong an inductive bias when the model is later fine-tuned on historical English. In any case, pre-training from scratch on historical data may be a more robust strategy than adapting a pre-trained model.

5 Conclusion & Future Work

Our experiments have shown the potential of historical pre-training for diachronically-relevant tasks. Historical pre-training, however, did not benefit the processing of historical texts from all different time spans to the same extent. A more balanced pre-training dataset could help alleviate these issues. Still, since collecting new data for certain time spans and genres is hindered by the scarcity of such material, researchers are left with the only option of up-sampling the available resources—c.f. Bamman and Burns (2020). The benefits of up-sampling for ranges of the diachrony that are lesser sourced should thus be explored.

Moreover, we have gained insight on the relative merit of different approaches to historical pre-training (pre-training from scratch vs. adapting a pre-existing model). This insight suggests a further experiment in which the "BERT-based Uncased" architecture is fine-tuned on the same dataset as `MacBERTh`, and the resulting model is put to test alongisde `MacBERTh` in order to see whether the claim holds true. However, considering the elevated cost of experimenting with MLM architectures, future research may want to refrain from costly practices like ablation studies and, instead, look at statistical modeling in order to find out the effect of particular design choices—e.g. can excessive sub-word tokenization be responsible for the drop in performance?

Finally, some of the benchmark tasks we implemented were designed ad-hoc to test the capabilities of MLMs at handling historical text. Future work should look into the deployment and evaluation of MLMs in real-word Humanities and DH scenarios in order to scale up the automated retrieval of otherwise difficult to access pieces of information. Besides fine-tuning on appropriate downstream tasks, current NLP research points towards "prompt engineering" (see Liu et al. (2021) for a recent survey) as a promising approach.

References

David Bamman and Patrick J Burns. 2020. Latin BERT: A contextual language model for classical philology.

David Bamman and Gregory Crane. 2011. Measuring historical word sense variation. In *Proceedings of the 11th annual international ACM/IEEE joint conference on Digital libraries*, pages 1–10.

Kaspar Beelen, Federico Nanni, Mariona Coll Ardanuy, Kasra Hosseini, Giorgia Tolfo, and Barbara McGillivray. 2021. When time makes sense: A historically-aware approach to targeted sense disambiguation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2751–2761. Association for Computational Linguistics.

Steven Bird. 2006. Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, pages 69–72.

Marcel Bollmann. 2019. A large-scale comparison of historical text normalization systems. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3885–3898, Minneapolis, Minnesota. Association for Computational Linguistics.

Emanuela Boros, Elvys Linhares Pontes, Luis Adrián Cabrera-Diego, Ahmed Hamdi, José Moreno, Nicolas Sidère, and Antoine Doucet. 2020. Robust named entity recognition and linking on historical multilingual documents. In *Conference and Labs of the Evaluation Forum (CLEF 2020)*, volume 2696, pages 1–17. CEUR-WS Working Notes.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Maud Ehrmann, Matteo Romanello, Alex Flückiger, and Simon Clematide. 2020. Overview of CLEF HIPE 2020: Named entity recognition and linking on historical newspapers. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, pages 288–310. Springer International Publishing.

Susan Fitzmaurice, Justyna A Robinson, Marc Alexander, Iona C Hine, Seth Mehl, and Fraser Dallachy. 2017. Linguistic dna: Investigating conceptual change in early modern english discourse. *Studia Neophilologica*, 89(sup1):21–38.

Lauren Fonteyn. 2020. What about grammar? Using BERT embeddings to explore functional-semantic shifts of semi-lexical and grammatical constructions. In *Proceedings of the Workshop on Computational Humanities Research (CHR 2020)*, volume 2723 of *CEUR Workshop Proceedings*, pages 257–268.

Mario Giulianelli, Marco Del Tredici, and Raquel Fernández. 2020. Analysing lexical semantic change with contextualised word representations. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 3960–3973, Online. Association for Computational Linguistics.

Edouard Grave. 2017. Language Identification · fastText.

Xiaochuang Han and Jacob Eisenstein. 2019. Unsupervised domain adaptation of contextualized embeddings for sequence labeling. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4238–4248, Hong Kong, China. Association for Computational Linguistics.

Yoonseok Heo, Sangwoo Kang, and Jungyun Seo. 2020. Hybrid sense classification method for large-scale word sense disambiguation. *IEEE Access*, 8:27247–27256.

Kasra Hosseini, Kaspar Beelen, Giovanni Colavizza, and Mariona Coll Ardanuy. 2021a. Neural Language Models for Nineteenth-Century English. *Journal of Open Humanities Data*, 7:22.

Kasra Hosseini, Kaspar Beelen, Giovanni Colavizza, and Mariona Coll Ardanuy. 2021b. Neural Language Models for Nineteenth-Century English (dataset; language model zoo).

Renfen Hu, Shen Li, and Shichen Liang. 2019. Diachronic sense modeling with deep contextualized word embeddings: An ecological view. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3899–3908, Florence, Italy. Association for Computational Linguistics.

Eric Huang, Richard Socher, Christopher Manning, and Andrew Ng. 2012. Improving word representations via global context and multiple word prototypes. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 873–882, Jeju Island, Korea. Association for Computational Linguistics.

Leonard Konle and Fotis Jannidis. 2020. Domain and Task Adaptive Pretraining for Language Models. In *Proceedings of the Workshop on Computational Humanities Research (CHR 2020)*, volume 2723 of *CEUR Workshop Proceedings*, pages 248–256.

Anthony Kroch, Beatrice Santorini, and Lauren Delfs. 2004. Penn-helsinki parsed corpus of early modern english.

British Library Labs. 2014. Digitised books. c. 1510 - c. 1900. json (ocr derived text).

Kai Labusch, Clemens Neudecker, and David Zellhöfer. 2019. Bert for named entity recognition in contemporary and historical german. In *Proceedings of the 15th Conference on Natural Language Processing, Erlangen, Germany*, pages 8–11.

Adam Langley and Dan S. Bloomberg. 2007. Google Books: making the public domain universally accessible. In *Document Recognition and Retrieval XIV*, volume 6500, pages 148 – 157. International Society for Optics and Photonics, SPIE.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*.

Enrique Manjavacas, Ákos Kádár, and Mike Kestemont. 2019. Improving lemmatization of non-standard languages with joint learning. In *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, volume 1, pages 1493–1503. Association for Computational Linguistics.

Jani Marjanen, Lidia Pivovarova, Elaine Zosa, and Jussi Kurunmaki. 2019. Clustering Ideological Terms in Historical Newspaper Data with Diachronic Word Embeddings. In *The 5th International Workshop on Computational History (HistoInformatics 2019)*, volume 2461 of *CEUR Workshop Proceedings*, pages 21–29.

Carlos Martinez-Ortiz, Tom Kenter, Melvin Wevers, Pim Huijnen, and Joris van Eijnatten. 2019. Design and implementation of ShiCo: Visualising shifting concepts over time. In *The 5th International Workshop on Computational History (HistoInformatics 2019)*, volume 1632 of *CEUR Workshop Proceedings*, pages 11–19.

Martin Mueller, Philip R Burns, and Craig A Berry. 2016. Collaborative curation and exploration of the eebo-tcp corpus. In Laura Estill, Diane K. Jakacki, and Michael Ullyot, editors, *Early Modern Studies after the Digital Turn*, chapter 7, pages 147–167. Iter and the Arizona Center for Medieval and Renaissance Studies.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Mohammad Taher Pilehvar and Jose Camacho-Collados. 2019. WiC: the word-in-context dataset for evaluating context-sensitive meaning representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1267–1273, Minneapolis, Minnesota. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Ville Satopaa, Jeannie Albrecht, David Irwin, and Barath Raghavan. 2011. Finding a "kneedle" in a haystack: Detecting knee points in system behavior. In *2011 31st International Conference on Distributed Computing Systems Workshops*, pages 166–171.

Dominik Schlechtweg, Barbara McGillivray, Simon Hengchen, Haim Dubossarsky, and Nina Tahmasebi. 2020. SemEval-2020 task 1: Unsupervised lexical semantic change detection. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1–23, Barcelona (online). International Committee for Computational Linguistics.

Mike Schuster and Kaisuke Nakajima. 2012. Japanese and korean voice search. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5149–5152. IEEE.

Stefan Schweter. 2020. Europeana bert and electra models.

Stefan Schweter and Johannes Baiter. 2019. Towards robust named entity recognition for historic German. In *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*, pages 96–103, Florence, Italy. Association for Computational Linguistics.

Stefan Schweter and Luisa März. 2020. Triple e-effective ensembling of embeddings and language models for ner of historical german. In *Conference and Labs of the Evaluation Forum (CLEF 2020)*, volume 2696. CEUR-WS Working Notes.

John Simpson and Edmund Weiner. 1989. *Oxford English Dictionary*. Oxford University Press.

Pia Sommerauer and Antske Fokkens. 2019. Conceptual Change and Distributional Semantic Models: An Exploratory Study on Pitfalls and Possibilities. In *Proceedings of the 1st International Workshop on Computational Approaches to Historical Language Change*, pages 223–233. Association for Computational Linguistics.

Nina Tahmasebi, Lars Borin, Adam Jatowt, et al. 2018. Survey of computational approaches to diachronic conceptual change. *arXiv preprint arXiv:1811.06278*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27.

A Appendix

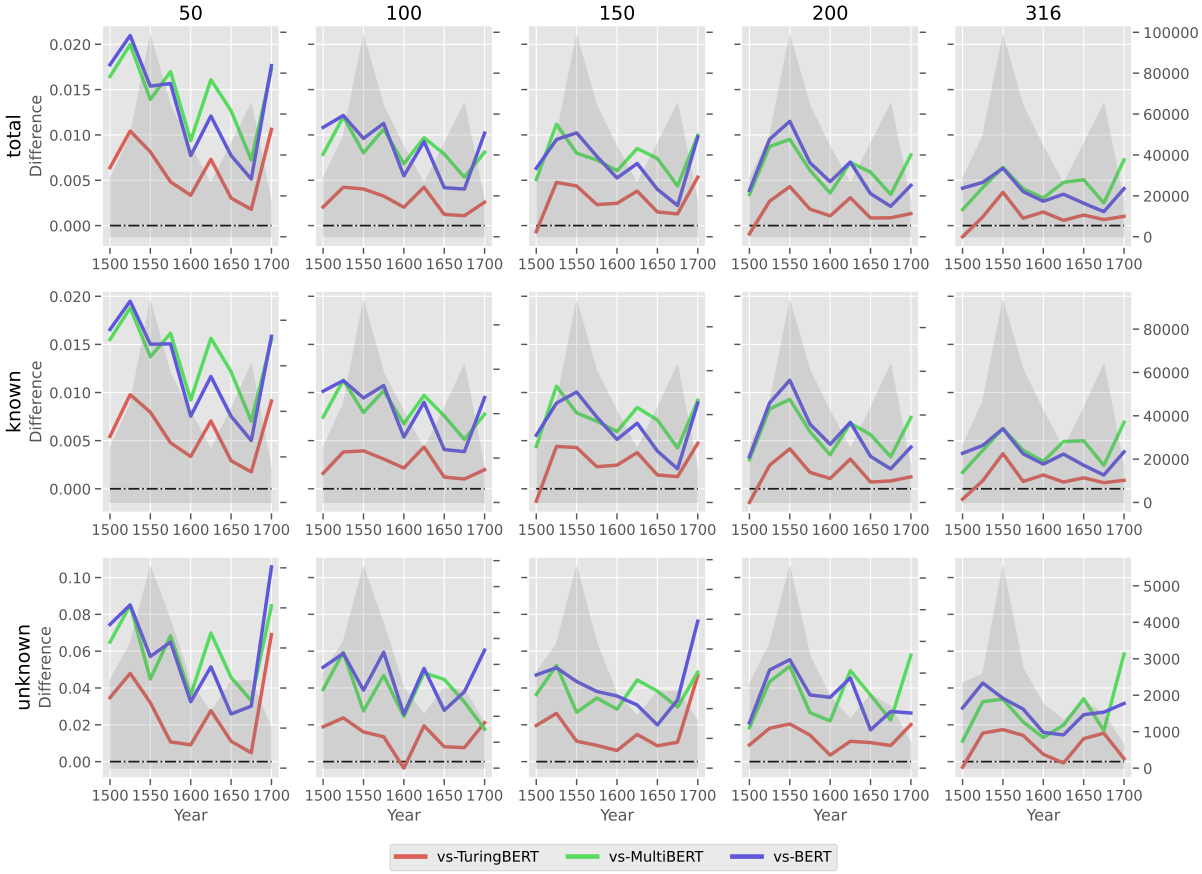

Figure 10: Difference in part-of-speech tagging accuracy of known, unknown and all tokens of `MacBERTh` with respect to the alternative models across training regimes.

	Left Quotation	Right Quotation
Example	He lov'd his Country with too unskilful a tenderness.	I love it to be grieved when he hideth his smiles.
Input	He [TGT] lov'd [TGT] his Country with too unskilful a tenderness.	I [TGT] love [TGT] it to be grieved when he hideth his smiles.
Sense	**1.a** *"To have or feel love towards (a person, a thing personified) (for a quality or attribute); to entertain a great affection, fondness, or regard for; to hold dear."*	**3.c** *"With direct object and infinitive or clause: to desire or like (something to be done). Also (chiefly U.S.) with for preceding the notional subject of the infinitive clause."*

Table 3: An example negative pair for lemma "love" showcasing the modification in order to fine-tune the model.

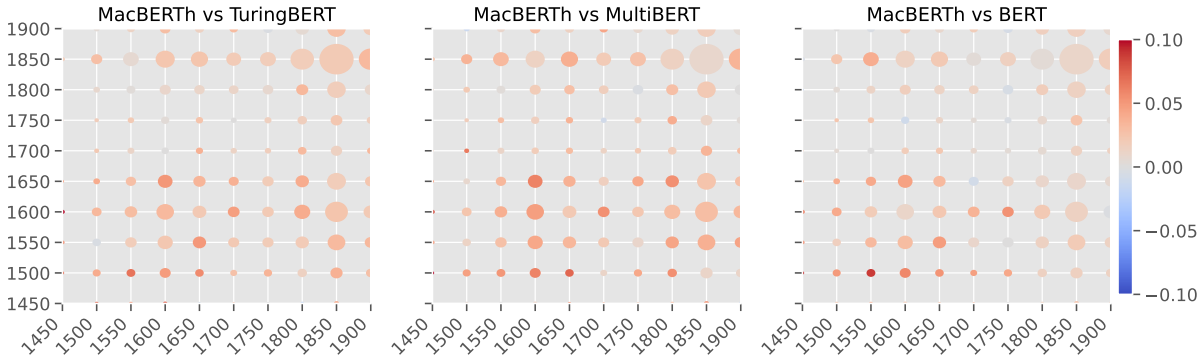

Figure 11: Comparison factoring in the periods of left (y-axis) and right (x-axis) input quotations. Each circle encodes the number (proportional to the radius) as well as the relative difference in accuracy with respect to `MacBERTh` (with red and blue respectively indicating whether `MacBERTh` out- or underperforms the compared models).

Named Entity Recognition for French medieval charters

Sergio Torres Aguilar
École nationale des chartes
sergio.torres@chartes.psl.eu

Dominique Stutzmann
CNRS-IRHT
dominique.stutzmann@irht.cnrs.fr

Abstract

This paper presents the process of annotating and modelling a corpus to automatically detect named entities in medieval charters in French. It introduces a new annotated corpus and a new system which outperforms state-of-the art libraries. Charters are legal documents and among the most important historical sources for medieval studies as they reflect economic and social dynamics as well as the evolution of literacy and writing practices. Automatic detection of named entities greatly improves the access to these unstructured texts and facilitates historical research. The experiments described here are based on a corpus encompassing about 500k words (1200 charters) coming from three charter collections of the 13th and 14th centuries. We annotated the corpus and then trained two state-of-the art NLP libraries for Named Entity Recognition (Spacy and Flair) and a custom neural model (Bi-LSTM-CRF). The evaluation shows that all three models achieve a high performance rate on the test set and a high generalization capacity against two external corpora unseen during training. This paper describes the corpus and the annotation model, and discusses the issues related to the linguistic processing of medieval French and formulaic discourse, so as to interpret the results within a larger historical perspective.

1 Introduction

Named entity recognition (NER) is a fundamental task aiming at detecting and classifying words used as rigid designators in a text. Typically the operation consists in a lexical segmentation of texts separating entities from common words and a subsequent classification of them according to a set of predefined categories. NER has quickly become part of the NLP toolbox used by digital humanities to structure and mine textual collections. However, its application to historical texts still involves some challenges. Not only medieval charters use low-resource languages such as medieval versions of Latin until the 15th century and vernacular languages (e.g. Old and Middle French) from the 13th c. onwards, but they are also written in diverse linguistic versions due to language change over space and time. Moreover their strong topic-dependency further complicates the use of general classifiers and popular NLP tools used for modern languages, since charters contain mainly property deeds whose wording was framed by well-defined documentary models using stereotyped structures and a restricted and formulaic vocabulary.

In the last years, digitizing and scholarly editing original manuscript sources have gained great momentum, coinciding with their reappraisal in medieval studies. New tools are needed to explore and mine these specialized sources in order to foster insights from millions of documents and support new hypotheses. Named entity recognition, in particular, is a key element in order to provide an indexed structure to historical texts. It would allow the implementation of information retrieval techniques and adapt diplomatics and historical research methods to large scale corpora.

Our contribution can be summarized as follows: (1) An annotated corpus built upon three different collections of medieval acts in French, (2) an adequate training and validation framework, to create supervised models able to automatically distinguish places and person names in unstructured texts; (3) We suggest a test protocol to evaluate the models' ability to generalize on a wide range of acts regardless of regional and chronological differences.

2 Related work

Sequence tagging, including NER, is a classic NLP task. NER processing libraries such as Spacy, Flair, and Stanford CoreNLP have become popular in digital humanities, because one can easily train custom models and apply diverse neural approaches based on RNN architectures using control gates as in the case of LSTM approaches or adding a double-scope lecture (Bi-LSTM) (Schmitt et al.,

Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH), pages 37–46
December 19, 2021.

2019). Indeed, LSTM architectures using statistical classifiers such as CRF and LDA have become the most successful training methods, esp. when feature engineering cannot be deployed as it happens with low-resources languages. While supervised corpus-based methods have generally become the paradigm in NER research, NER approaches on ancient Western languages as classical Latin and Greek are still deploying ruled-based analyzers coupled with gazetteers and patronymic lists (Erdmann et al., 2016; Milanova et al., 2019) due to the lack of relevant annotated corpora. Moreover, research on NER for heritage resources focused on 19th century OCRed newspaper collections and it has not been confronted with the intense spelling variations of pre-modern, so-called "pre-orthographic" sources (Ehrmann et al., 2016; Kettunen et al., 2016).

Currently there are only few available NLP resources for pre-modern French: a dependency Treebank, the Syntactic Reference Corpus of Medieval French (Prévost and Stein, 2013), and two lemmatizers for Old French, one trained on the Base de Français Médiéval (Guillot et al., 2018) using TreeTagger and the other, Deucalion, based on a Encoder-Decoder architecture (Clérice and Camps, 2021). But there is a lack of large language models for tasks such as topic clustering and named entity recognition, given that PoS tools only detect, but do not classify proper names or deal with their length and composition.

(Zhang et al., 2020) show that robust NER models can be trained even in the absence of word-level features (e.g. lemma, POS), but that one then has to add character-based word representations to encode all lexical phenomena, concatenated with word embeddings vectors. Leveraging on pre-trained language models increase significantly the performance compared with traditional approaches (Ehrmann et al., 2021). Yet, static embeddings and contextualized word representations such as BERT (Devlin et al., 2018) and ELMO (Peters et al., 2018), require large-scale corpora for training and fine-tuning, and they are not available for ancient language versions ("*état de langue*") or domain-specific texts.

3 Corpus description

To remedy the lack of relevant training corpora, we created a relatively large dataset for the present task, composed of ca. 500,000 words, from three different sources: *Diplomata Belgica*, *HOME-Alcar*, and the *Arbois (CBMA)*.

3.1 Diplomata Belgica (DiBe)

The *Diplomata Belgica* [1] are a large database published by the Belgian Royal Historical Commission in 2014. It contains more than 35,000 critical references and almost 19,000 full transcriptions of mostly Latin and middle French charters (de Hemptinne et al., 2015). It is based on (Wauters and Halkin, 1866-1907; Bormans et al., 1907-1966). The edited charters range from the early 8th century (mostly royal diplomas) to the late 13th century with a high concentration on the final period from the mid-12th century (84% of the corpus). They are related to private and public business and issued by or for institutions and persons in nowadays Belgium and Northern France.

For this work, we have annotated 922 charters in French edited in the *Diplomata Belgica*. They all are dated in the 13th century, and transmit diverse legal actions (donations, privileges, concessions and confirmations, judicial sentences, sales and exchanges) concerning individuals and corporate bodies (lay or religious institutions). The main producers are: (1) aldermens (*échevins*) of Tournai, Arras and Cambrai, for 135 acts concerning mostly private business; (2) counts or countesses of Flanders, Hainaut, Laon, Bar, etc. for 164 acts, mostly for notifications and confirmations; (3) feudal lords for 214 acts, mostly donations and exchanges with abbeys, but also private business and charters of franchise for cities; (4) aldermen of Ypres, for 374 chirographs (i.e. charters produced in double or triple copy to give one to each stakeholder) written in the last third of the 13th century and concerning private affairs linked to trade and industry, e.g. sales, exchange contracts, loans, recognition of debts (Valeriola, 2019).

3.2 HOME-Alcar

The HOME-Alcar corpus published in 2021 (Stutzmann et al., 2021) was produced as part of the European research project *HOME History of Medieval Europe*, under the coordination of Institut de Recherche et d'Histoire des Textes (IRHT-CNRS). This corpus provides the images of medieval manuscripts aligned with their scholarly editions at line level as well as a complete annotation of named entities (persons and places), as a resource to train Handwritten Text Recognition

[1] https://www.diplomata-belgica.be/

	DiBe		Navarre		Fervaques		Saint_Denis		Arbois	
Acts (1190)	922		96		54		53		65	
Tokens (500 767)	311002		82878		24843		34275		47769	
category/ length	PERS	LOC	PERS	LOC	PERS	LOC	PERS	LOC	PERS	LOC
1	3570 (38%)	8895 (89%)	534 (48%)	1834 (94%)	241 (65%)	589 (90%)	266 (42%)	498 (72%)	368 (33%)	1353 (95%)
2	2156 (23%)	789 (8%)	130 (12%)	70 (4%)	34 (9%)	41 (6%)	123 (19%)	119 (17%)	218 (20%)	25 (2%)
3	3254 (34%)	93 (1%)	379 (34%)	23 (1%)	74 (20%)	29 (4%)	206 (32%)	29 (4%)	397 (36%)	25 (2%)
>3	504 (5%)	225 (2%)	62 (6%)	27 (1%)	22 (6%)	2 (0.3%)	43 (7%)	45 (7%)	126 (11%)	18 (1%)
# entities	9484	10001	1105	1954	371	661	638	691	1109	1421
# tokens	19722	11670	2148	2138	635	766	1320	1011	2544	1549
Density	6.32 %	3.74 %	2.59 %	2.58 %	2.55 %	3.08 %	3.85 %	2.94 %	5.32 %	3.24 %
Normalized	9.12 %		4.72 %		4.75 %		6.29 %		7.66 %	

Table 1: Statistics on entities for each corpus according to their length. *Density* represents the percentage of tokens belonging to entities. *Normalized* expresses the sum of densities without taking in account the nested LOC cases, v.g. the locative in a person name.

(HTR) and NER models.

HOME-Alcar includes 17 cartularies, i.e. volumes containing copies of charters, produced by religious or public institutions to keep a memorial record of his properties and rights. These cartularies were produced between the 12th and 14th centuries. The corpus contains 3090 acts, with 2760 in Latin and 330 in Old and Middle French, and almost 1M tokens. Texts in French can be found in 12 of the 17 cartularies, but only in three they constitute a substantial part that is adequate to train NER models: (1) Cartulary of Charles II of Navarre : 96 acts (Lamazou-Duplan et al., 2010); (2) Cartulary of Fervaques abbey : 54 acts (Schabel and Friedman, 2020); so-called "White Cartulary" of Saint-Denis Abbey : 53 acts (Guyotjeannin, 2019).

The first one is from a lay family. The transcribed acts, dated between the 1297 and 1372, contain private donations and exchanges as well as other legal categories that are uncommon in religious cartularies, e.g. treatises, successions, indemnities. The other cartularies were produced by religious institutions, namely Norman and Ile-de-France abbeys respectively, and contain mostly donations from private persons and privileges from public authorities. The French acts are dated between 1250 and 1285 for Fervaques and between 1244 and 1300 for Saint-Denis.

3.3 Arbois (CBMA)

The cartulary of the city of Arbois in the Jura region was written in 1384, but largely keeps the language of the earlier originals, even of the 13th c. The edition contains 50 acts plus 32 acts in annex (65 in French, 17 in Latin) (Stouff, 1989). They are included in the Corpus Burgundiae Medii Aevi (CBMA, # 11424-11506)[2], which provides the base text for our annotation.

Arbois was part of the Burgundian county and obtained a charter of franchises in 1247. The community, though under the seigneurial regime, was recognised as a corporate entity and able to trade land and become an owner. The acts of the cartulary show its economical and social interactions with the lords or other communities: agreements about public issues such as military services and war costs, or about taxes and customs; charters declaring communal land purchases or lawsuits in court; even accounts that show the financial problems of the community due to the expenses of fortifications and wars. The community was ruled and represented by the aldermens (*prud'hommes*) who probably commissioned the redaction of the cartulary.

4 Corpus annotation

4.1 Annotation parameters

Since the early 11th century, personal names start adopting the name and by-name structure. They are composed by a baptism first name and a second part that can be a personal surname, a patronymic name (*nomen paternum*), or a locative. The latter form makes up to a third of all place entities and provides precious historical information as they typically correspond to micro-toponyms, whose existence is often not recorded otherwise. The annotation is focused on the proper name acting as a rigid designator and does not include co-occurrences as personal titles, dignities or functions. For example in the named entity expression: "Philippe, par la grace de dieu, roy de France" we annotate "Philippe" (PERS) and "France" (LOC), but not the full entity.

The annotation only records person and place names. Names of corporate bodies entities have

[2] http://www.cbma-project.eu/

TOKEN	PERS	LOC	TOKEN	PERS	LOC
Jehan	B-PERS	O	Vigilie	O	O
de	I-PERS	O	Nostre	O	O
Le	I-PERS	B-LOC	Dame	O	O
Capelle	I-PERS	I-LOC	Candeleir	O	O
Jehain	B-PERS	O	tous	O	O
chastelain	O	O	li	O	O
de	O	O	capitele	O	O
Cambray	O	B-LOC	de	O	O
et	O	O	Notre	O	B-LOC
seigneur	O	O	Dame	O	I-LOC
d'	O	O	de	O	I-LOC
Oisy	O	B-LOC	Cambray	O	I-LOC
Estienne	B-PERS	O	Margrite	B-PERS	O
Le	I-PERS	O	veve	O	O
Lonbart	I-PERS	O	Watier	B-PERS	O
Adan	B-PERS	O	sans	I-PERS	O
Bridoul	I-PERS	O	Paour	I-PERS	O
Huon	B-PERS	O	et	O	O
Le	I-PERS	O	Selie	B-PERS	O
Fevre	I-PERS	O	,	O	O
Jehan	B-PERS	O	se	O	O
Wilame	I-PERS	O	fille	O	O

Table 2: Example of annotations for named entities in DiBe # 15541, # 16356, # 17169, and # 36741

been annotated as organisations (ORG) in *Diplomata Belgica* first, but then folded to "places" (LOC) as in the other corpora, because these entities are mostly ambiguous in medieval texts (the church of "Notre Dame" or the lordship of "Oisy" mean a place and a corporate body at the same time).

4.2 Annotation process

The charters of the HOME-Alcar corpus were already annotated following a double scope: flat entities (proper names and simple periphrasis) and full entities (proper names and co-occurrences). This annotation was made on the basis of an automatic annotation using a multilingual NER model, then later corrected by two expert annotators. Inter-annotator agreement was not measured, as corrections and ambiguous cases were discussed among the annotators during the process.

The charters of *Diplomata Belgica* and Arbois charters were annotated in the flat style in the same manner. We first applied an automatic multilingual model and a single expert manually corrected the hypothesis.

We use the usual BIO format to encode the annotated labels as follows: B-tag, I-tag and O-tag to represent Begin (B) of label, continuation (I) of label and absence (O), respectively.

5 Training of the models

5.1 Data preparation

Our gold-standard (ground-truth) corpus is composed of 1190 acts ($\sim$ 0,5 M tokens), divided into two sets in order to conduct two experiments: (1) training and test on a homogeneous corpus; (2) test on additional, external corpora to measure the robustness of the model.

The first experiment is based on a corpus containing 1072 documents and encompassing the *Diplomata Belgica* and the cartularies of Navarre and Fervaques. It is randomly split with a 0.8-0.2 ratio: training set (844 documents), and validation and test sets (45 and 183 documents). The results of the first experiment are shown in Table 3.

The second experiment uses a corpus composed of two corpora unseen during the first experiment, the cartularies of Saint-Denis and Arbois (118 documents). The classifiers trained on the entire first corpus were applied on the second. The results are shown in tables 4 and 5.

For training we consider each charter as one training unit with a max length of 3,000 words (and a median of 276) and a max word length of 12 characters (and a median of 5).

5.2 Problem definition

We see our problem as a traditional two-step sequence labeling task. The input is a defined sequence of tokens $x = (x_1, x_2...x_{n-1}, x_n)$ and the output must be defined as a sequence of tokens labels $y = (y_1, y_2...y_{n-1}, y_n)$.

Both steps (PERS and LOC) may be combined, successive or separate. In our implementations, Flair and Spacy have independent annotation processes and our custom model has two successive steps.

5.3 The custom Bi-LSTM-CRF model

In the first step we extract word and sub-word features using NLP tools; the second step involves the training of the neural classifiers. For the custom Bi-LSTM model this step occurs in two stages. First, we apply the classifier to produce the places names hypotheses. Then, the classifier integrates the hypotheses of place names as an extra feature and predicts the person names.

5.3.1 Model Architecture

As is shown in Figure 1 we train three embedding vectors from our data. First is a word represen-

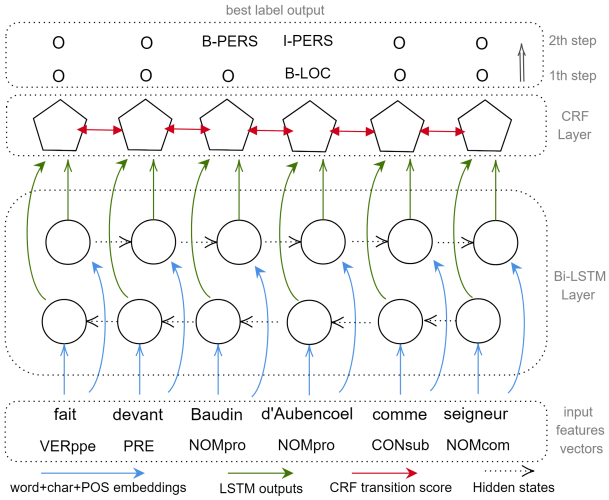

Figure 1: BiLSTM-CRF architecture using words, characters and POS embeddings as input on the excerpt "fait devant Baudin d'Aubencoel comme seigneur" (translates as: "written in front of Baudin d'Aubencoel acting as lord") tagged as a nested entity.

tation, second is a character-level representation, third is the POS character-level feature. Then, we merge these embeddings in order to form one single enriched vector and feed it into the Bi-LSTM. Finally the output hidden states are decoded using a CRF layer.

5.3.2 Character representations

Feeding the model with character-word information is a crucial step working with non-standard languages as it exploit sub-word level information as prefix and declension. In charters, many writing phenomena are linked to character variations due to the lack of grammatical rules or the introduction of spelling variations during redaction. In the same way, character-level representation naturally handles out-of-vocabulary as false lemmas, hapax and abbreviations, which can not be correctly expressed through word-vectors depending of a predefined dictionary and grammar pattern. Thus, character-level approaches are able to encode all textual phenomena at a morpheme-level using a restricted dictionary (115 keys in our work).

5.3.3 Word embeddings

Word embeddings improve the sequence tagging models. Yet, there are no publicly available embeddings for Middle French. In order to use pretrained embeddings we have trained a customized 300-dimensions and 44k vocabulary size word2vec model using a limited collections of medieval French charters (1.5M of tokens) which includes all the sub-corpora used in the present work plus French charters coming from other corpora as the CBMA and Île-de-France cartularies (Guyotjeannin, 2006-2010) [3]. Thus, the one-hot encoding words are replaced with their corresponding vectors.

5.3.4 POS information

The character-level embeddings do not catch morpho-syntactic information. To remedy this situation, POS features may be a workaround, specially working with large dependency plots as they import contextual features. But POS tags must be distributed among characters for each word. In this case inspired by the work of (Li et al., 2018), we generate a new feature combining character positions and POS tags. Positions are distributed using a 4-set tags as follows: B:Begin, M:Middle, E:end, S:single. The POS-tags were obtained using the TreeTagger lemmatizer made public by the Syntactic Reference Corpus of Medieval French in 2013 (Prévost and Stein, 2013).

5.3.5 Bi-LSTM-CRF Layer

Bidirectional long short-term memory (BiLSTM) models have proven to be effective for multiple sequence labelling tasks. As a classical RNN, the LSTM make output predictions based on long distance features using history information cells. The idea of the bidirectional variant is to reinforce the learning connecting the present and the past contexts of each token in the sentence. Thus, the output is a vector formed by the concatenation of a double sequence of LSTM hidden states $y_t = \overrightarrow{h_t} + \overleftarrow{h_t}$ for each token and token-features embeddings. This output is finally decoded by a Conditional Random Fields (CRF) layer which estimates the transition probabilities between tags and can predict the entire label sequence in each time step.

5.3.6 Training hyper-parameters

The grid search was evaluated on four key options: *batch-size* $\in$ { 2, 4, 16, 32 }, *output embeddings dimensions* $\in$ { 100, 200, 400 }, *learning methods* $\in$ { sgd, adam, rmsprop }, and *dropout* $\in$ { 0.2, 0.3, 0.4 } . Optimal combination was chosen following a 4-batch size, 200-dimensions embeddings, 0.2 dropout and rmsprop optimizer using a ReduceLROnPlateau scheduler.

[3] http://elec.enc.sorbonne.fr/

(a)

Model/ category	Flair			Spacy			Custom			Support
	Pr	Rc	F1	Pr	Rc	F1	Pr	Rc	F1	
B-PERS	0.938	0.959	0.949	0.948	0.969	0.958	0.975	0.982	**0.978**	2128
I-PERS	0.988	0.983	**0.986**	0.978	0.980	0.979	0.961	0.984	0.973	1805
micro avg	0.961	0.970	0.966	0.962	0.974	0.968	0.969	0.983	**0.977**	3933
B-LOC	0.960	0.945	0.952	0.956	0.952	0.954	0.967	0.975	**0.971**	2333
I-LOC	0.926	0.872	0.898	0.945	0.900	**0.922**	0.926	0.913	0.919	360
micro avg	0.956	0.935	0.945	0.954	0.945	0.950	0.961	0.966	**0.964**	2693

(b)

	PERS	LOC	PERS	LOC	PERS	LOC
Correct (TP)	2032	2187	2047	2213	2050	2210
Partial	16	27	26	24	47	33
Missed (FN)	80	119	55	96	31	90
Spurius (FP)	127	82	102	88	49	49
Pr	0.934	0.952	0.941	0.952	**0.955**	**0.964**
Rc	0.954	0.937	0.962	**0.948**	**0.963**	0.947
F1	0.944	0.945	0.951	0.950	**0.959**	**0.956**

Table 3: Evaluation results on test set for Flair, Spacy and Custom NER taggers : Pr (Precision), Rc (Recall), F1 (F1 score), TP (True positive), FN (False negative), FP (False positive), micro avg (micro-averaging score), Support (number of observations). First table (a) indicates tag-level performance; second table (b) indicates entity-level performance.

5.4 The SpaCy model

Spacy is an open-source NLP library proposing transformer-based pipelines. We fit our dataset on the default SpaCy NER architecture (Honnibal and Montani, 2017) which rely on two sequential encoding + attention networks: the first one is a typical embedding encoder which transforms tokens into a continuous vector space; the second one applies a manual extraction-features mechanism from the encoded tokens, evaluating the connections between tokens in a similar way of an attention layer. The goal is to give to each word a unique representation for each of its different contexts. Finally, a Multilayer Perceptron outputs the entity label. For the encoding step, we load in Spacy as backbone the transformers library CamemBert (Martin et al., 2019) trained on French modern texts.

For the Spacy model, as the default architecture does not accept other NER features, we trained two separate models for places and persons.

5.5 The Flair model

Flair is a PyTorch based NLP library which achieves state-of-art performance using pre-trained contextual embeddings as ELMO and BERT. The Flair NER architecture uses a deep learning architecture with Bi-LSTM layers in the back-end and allows users to activate CRF taggers (Akbik et al., 2018). In our case we deploy a special feature called "stacked embeddings" which combines classic embeddings with contextual embeddings in one single Pytorch vector. In our modelling we used the FastText (Bojanowski et al., 2017) embeddings trained on French Wikipedia as word-vectors combined to the default Flair multilingual forward + backward contextual embeddings. As for Spacy, we trained two separate Flair models for places and persons.

6 Evaluation

Table 3 shows the best results obtained with a training set of 1 072 charters. We provide the usual Precision, Recall and F1-score metrics at a token-level (B- and I- tags). We also include full-entity level metrics on strict match: strict match occurs when the hypothesis and the ground-truth match perfectly.

All three models obtain high performance results, both in PERS (0.944 to 0.959) and LOC (0.945 to 0.956) categories. The first metric shows that performance between B- and I- tags are harmonic which implies that our three models are able to correctly detect the boundaries of the entities regardless of their length. The second metric confirms that false negatives and false positives are marginal both in LOC and PERS thus achieving a very good result in multi-class tasks.

The custom Bi-LSTM shows a better performance in most categories. Specifically it generalizes better as indicated by the lower number of false negatives and false positives compared to the Flair and Spacy models. It can be explained by the use of a denser set of features including PoS and French medieval embeddings more adapted to the medieval charters. However, the Flair and Spacy performance is only 1 to 2 points lower, confirming

	Saint-Denis									
	Model/	Flair			Spacy			Custom			Support
	category	Pr	Rc	F1	Pr	Rc	F1	Pr	Rc	F1	
	B-PERS	0.959	0.964	0.962	0.954	0.970	0.962	0.982	0.969	**0.975**	638
	I-PERS	0.955	0.930	0.942	0.945	0.959	0.952	0.975	0.961	**0.968**	682
(a)	micro avg	0.957	0.946	0.952	0.962	0.974	0.968	0.978	0.975	**0.976**	1320
	B-LOC	0.898	0.916	0.907	0.905	0.926	0.915	0.926	0.942	**0.934**	691
	I-LOC	0.885	0.863	0.873	0.899	0.855	0.877	0.904	0.873	**0.889**	320
	micro avg	0.894	0.899	0.896	0.903	0.903	0.903	0.919	0.920	**0.920**	1011
		PERS	LOC		PERS	LOC		PERS	LOC		
	Correct (TP)	596	627		598	633		597	638		
	Partial	19	29		21	29		27	36		
(b)	Missed (FN)	23	35		19	29		14	17		
	Spurius (FP)	26	49		29	41		9	38		
	Pr	0.930	0.890		0.922	**0.900**		**0.943**	0.896		
	Rc	0.934	0.907		**0.937**	0.916		0.936	**0.923**		
	F1	0.932	0.898		0.930	0.908		**0.939**	**0.909**		

Table 4: Results of model evaluation on the cartulary of Saint Denis.

	Arbois									
	Model/	Flair			Spacy			Custom			Support
	category	Pr	Rc	F1	Pr	Rc	F1	Pr	Rc	F1	
	B-PERS	0.945	0.920	0.932	0.933	0.944	0.938	0.922	0.953	**0.938**	1109
	I-PERS	0.940	0.954	**0.947**	0.942	0.944	0.943	0.966	0.890	0.927	1435
(a)	micro avg	0.942	0.939	0.941	0.938	0.944	**0.941**	0.947	0.917	0.932	2544
	B-LOC	0.936	0.939	0.938	0.927	0.953	0.940	0.923	0.968	**0.945**	1421
	I-LOC	0.746	0.758	0.752	0.773	0.791	0.782	0.837	0.798	**0.817**	128
	micro avg	0.920	0.924	0.922	0.915	0.939	0.927	0.916	0.954	**0.934**	1549
		PERS	LOC		PERS	LOC		PERS	LOC		
	Correct (TP)	982	1326		998	1350		979	1362		
	Partial	50	28		57	18		95	20		
(b)	Missed (FN)	77	67		54	53		35	39		
	Spurius (FP)	47	73		62	91		76	108		
	Pr	**0.910**	**0.929**		0.893	0.925		0.851	0.914		
	Rc	0.885	0.933		**0.900**	0.950		0.883	**0.958**		
	F1	**0.897**	0.931		0.897	**0.937**		0.867	0.936		

Table 5: Results of model evaluation on the cartulary of Arbois.

that a robust model for the recognition of nested entities can be trained on our corpus using domain unspecific embeddings or without additional linguistic features, which are not usually available for ancient language versions.

In general, the three models perform slightly less well on I-LOC suggesting some issues on LOC compound entities. Average results are not affected severely since only 5-10% of the entities belongs to this type (see table 1). A close inspection reveals that I-LOC errors are propagated due to the presence of some uncommon and large entities (*Saint Nicolay des Pres en costé Tournay*, *Jehan Godin de Maisieres sur Meuse*) in which the models make a partial or a double annotation. In other cases, errors are triggered for LOC or PERS by the presence of large periphrastic denominations (*Margherite suer Abreham Bertelot veve Jehan le Crudenere*) or for less common entity particles as the article "li" (*Jehans li Cornus del Fontenil*) or the Flemish preposition *van* (*Watier van Wormezeele*). In some other cases, particles associated to regular vocabulary as *des*, *d'*, *de la*, *dou* which can be connected to multiple contexts, may induce a partial match (*Willaumes dou Temple li Macheclers*, *Saint Johan du Pié des Pors*).

6.1 Evaluation on external corpora

Tables 4 and 5 show that the results obtained on the test set are largely replicated when the models are applied to external corpora (Arbois and Saint Denis), with a harmonic precision and recall for the three models, a similar performance in B- and I-tags and a high performance in strict match ranging from 0.897 to 0.939 in PERS and from 0.898 to 0.937 in LOC, thus confirming that generalization in external documents only causes a small drop in performance (2 to 3 points). Again the custom

Bi-LSTM model seems to perform better than the Flair and Spacy models.

In general, charters from Saint-Denis are closer in style to the ones in *Diplomata Belgica* and Fervaques because they record similar legal actions and uses similar formulas. The performance is slightly lower in Arbois which is a municipal cartulary with a later chronology registering many juridical actions rarely found in the training corpus.

As in the test set, the main problem remains the recognition on the I-LOC tag. The same kind of errors appears as in the first experiment, but most are triggered by festivity names, a type that had not been annotated. In charters, dates are indicated using the saints' festivities and appear using a saint's name which was learned as a LOC entity by the model (*a paier le ior Saint Martin*, *chascun an es huiteves Saint Denys*). In consequence in some cases the models may propose a false positive.

7 Discussion

This work clearly proves that robust tools to classify named entities on French medieval charters can be modeled using character-based neural network approaches (Bi-LSTM-CRF) on our annotated corpus. The test sets being of different dates (end of 13th c. and end of 14th c.), the high performance proves that the models are fairly robust against language change. This may be explained as follows.

1) In Northern France, during the late Middle Ages, the anthroponymic structure is stable. Not only the stock of first names is limited (e.g. in *Diplomata Belgica*, 47% of persons use one of the ten top names *Jehans, Watiers, Jakemes, Willaumes, Henris, Pieres, Nicholes, Bauduins, Gilles, Margherite*, and their variants), but also by-name and even the periphrastic denomination follow a recurrent pattern in which the model easily fits.

2) In a similar way, the named entity co-occurrences, which are crucial to calculate transition scores, belongs to a restricted stock. In charters there are a constant reference to a well-delineated territorial space as well as to broad system of titles, offices and dignities presenting a person. For example, in *Diplomata Belgica*, five terms (*sire/messire, bourgois, signor/monsigneur, eschevin, dame/madame*) co-occur in 24% of all personal entities.

3) Moreover, the lexical and semantic contexts of appearance of the named entities are well-defined by the use of formulaic models. Formulas are not fixed and charters are not mass-produced nor standardized, but they involve the use of a restricted vocabulary and are constrained by their need to follow a certain form, since they are documents with legal value.

These circumstances greatly help to obtain a valid NER model starting from a limited collection of charters. We have demonstrated that custom and off-the-shelf library models are able to capture the underlying structure of the charters' entities even using a small set of features and can be successfully applied to other diplomatic collections in spite of chronological and regional differences. Most errors concern partial matches on untypical data or complex data on which the model fits hardly because certain lexical series are missed or hindered.

8 Conclusion

We present an annotated corpus to French medieval charters and three neural NER models. The evaluation returns a strong performance reaching 0.96 in both PERS and LOC categories on the homogeneous test set and 0.95 in PERS and 0.92 in LOC on unseen data which confirms that the model can be used on charters from other chronologies and origins.

Besides, we can confirm that our models are able to produce a double hypothesis which implies a high confidence on the recognition of nested entities extensively used in medieval charters.

While this work concerns the development of a neural NER model, it can benefit several research areas including indexation systems, data visualization and distant reading methods. Named entities are the base of several digital and classical humanities research methods on networks, timelines, event-lines and GIS-maps. These models and the annotated data on which it is built, which are themselves new contributions, can be easily integrated into other pipelines, thus contributing to enhance the toolbox for Old and Middle French regarding other supervised methods.

9 Model repositories

The models, source code and the annotated corpora supporting this work are available at (Torres Aguilar, 2021)

References

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th international conference on computational linguistics*, pages 1638–1649.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Stanislas Bormans, Fabienne Marien, Joseph Halkin, J. Cuvelier, Jean-Jacques Hoebanx, and Charles Wirtz. 1907-1966. *Table chronologique des chartes et diplômes imprimés concernant l'histoire de la Belgique*. Commission royale d'histoire, Palais des Académies, Bruxelles.

Thibault Clérice and Jean-Baptiste Camps. 2021. chartes/deucalion-model-af: 0.4.0Alpha.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Maud Ehrmann, Giovanni Colavizza, Yannick Rochat, and Frédéric Kaplan. 2016. Diachronic evaluation of ner systems on old newspapers. In *Proceedings of the 13th Conference on Natural Language Processing (KONVENS 2016)*, CONF, pages 97–107. Bochumer Linguistische Arbeitsberichte.

Maud Ehrmann, Ahmed Hamdi, Elvys Linhares Pontes, Matteo Romanello, and Antoine Doucet. 2021. Named entity recognition and classification on historical documents: A survey. *arXiv preprint arXiv:2109.11406*.

Alex Erdmann, Christopher Brown, Brian D Joseph, Mark Janse, Petra Ajaka, Micha Elsner, and Marie-Catherine de Marneffe. 2016. Challenges and solutions for latin named entity recognition. In *COLING 2016: 26th International Conference on Computational Linguistics*, pages 85–93. Association for Computational Linguistics.

Céline Guillot, Serge Heiden, and Alexei Lavrentiev. 2018. Base de français médiéval: une base de référence de sources médiévales ouverte et libre au service de la communauté scientifique. *Diachroniques. Revue de Linguistique française diachronique*, (7):168–184.

Olivier Guyotjeannin. 2006-2010. Cartulaires numérisés d'ile-de-france.

Olivier Guyotjeannin. 2019. Chartes de l'abbaye de Saint-Denis.

Thérèse de Hemptinne, Jeroen Deploige, Jean-Louis Kupper, and Walter Prevenier. 2015. Diplomata belgica: les sources diplomatiques des pays-bas méridionaux au moyen âge. the diplomatic sources from the medieval southern low countries.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Kimmo Kettunen, Eetu Mäkelä, Teemu Ruokolainen, Juha Kuokkala, and Laura Löfberg. 2016. Old content and modern tools-searching named entities in a finnish ocred historical newspaper collection 1771-1910. *arXiv preprint arXiv:1611.02839*.

Véronique Lamazou-Duplan, Anne Goulet, and Philippe Charon. 2010. *Le cartulaire dit de Charles Ii roi de Navarre*. Presses universitaires de Pau et des Pays de l'Adour.

Yanzeng Li, Tingwen Liu, Diying Li, Quangang Li, Jinqiao Shi, and Yanqiu Wang. 2018. Character-based bilstm-crf incorporating pos and dictionaries for chinese opinion target extraction. In *Asian Conference on Machine Learning*, pages 518–533.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric Villemonte de La Clergerie, Djamé Seddah, and Benoît Sagot. 2019. Camembert: a tasty french language model. *arXiv preprint arXiv:1911.03894*.

Ivona Milanova, Jurij Silc, Miha Serucnik, Tome Eftimov, and Hristijan Gjoreski. 2019. Locale: A rule-based location named-entity recognition method for latin text. In *HistoInformatics@ TPDL*, pages 13–20.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Sophie Prévost and Achim Stein. 2013. Syntactic reference corpus of medieval french (srcmf). *Lyon & Stuttgart: ENS de Lyon*.

Chris Schabel and Russell L. Friedman. 2020. *The Cartulary of Fervaques Abbey, a Cistercian Nunnery*. in press.

Xavier Schmitt, Sylvain Kubler, Jérémy Robert, Mike Papadakis, and Yves LeTraon. 2019. A replicable comparison study of ner software: Stanfordnlp, nltk, opennlp, spacy, gate. In *2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS)*, pages 338–343. IEEE.

Louis Stouff. 1989. *Cartulaire de la ville d'Arbois au comté de Bourgogne*. Revue bourguignonne de l'enseignement supérieur, 8, n° 2.

Dominique Stutzmann, Sergio Torres Aguilar, and Paul Chaffenet. 2021. HOME-Alcar: Aligned and Annotated Cartularies. Type: dataset.

Sergio Torres Aguilar. 2021. Named Entity Recognition for French medieval charters. Models and datasets.

Sébastien de Valeriola. 2019. Le corpus des chirographes yprois, témoin essentiel d'un réseau de crédit du xiiie siècle. *Bulletin de la Commission royale d'Histoire*, 185(1):5–74.

Alphonse Wauters and J. Halkin. 1866-1907. *Table chronologique des chartes et diplômes imprimés concernant l'histoire de la Belgique*. M. Hayez, Bruxelles.

Yu Zhang, Zhenghua Li, Houquan Zhou, and Min Zhang. 2020. Is pos tagging necessary or even helpful for neural dependency parsing?

Processing M.A. Castrén's Materials: Multilingual Typed and Handwritten Manuscripts

Niko Partanen[0000-0001-8584-3880]
Department of Finnish, Finno-Ugrian and Scandinavian Studies
University of Helsinki

Jack Rueter[0000-0002-3076-7929]
Department of Digital Humanities
University of Helsinki

Mika Hämäläinen[0000-0001-9315-1278]
Department of Digital Humanities
University of Helsinki & Rootroo Ltd

Khalid Alnajjar[0000-0002-7986-2994]
Department of Digital Humanities
University of Helsinki & Rootroo Ltd

firstname.surname@helsinki.fi

Abstract

The study forms a technical report of various tasks that have been performed on the materials collected and published by Finnish ethnographer and linguist, Matthias Alexander Castrén (1813–1852). The Finno-Ugrian Society is publishing Castrén's manuscripts as new critical and digital editions, and at the same time different research groups have also paid attention to these materials. We discuss the workflows and technical infrastructure used, and consider how datasets that benefit different computational tasks could be created to further improve the usability of these materials, and also to aid the further processing of similar archived collections. We specifically focus on the parts of the collections that are processed in a way that improves their usability in more technical applications, complementing the earlier work on the cultural and linguistic aspects of these materials. Most of these datasets are openly available in Zenodo. The study points to specific areas where further research is needed, and provides benchmarks for text recognition tasks.

1 Introduction

As a research domain, the Natural Language Processing has regularly focused on the formal written varieties of the most widely used languages of the world. At the same time there has been a growing interest in both non-standard and informal language (Hämäläinen et al., 2021; Partanen et al., 2019), and their historical varieties (Säily et al., 2021; Partanen et al., 2021). The research potential of historical language varieties is clearly on the upbound, and one can argue that the need is already quite evident, as digitization processes in libraries and archives around the world have reached relatively mature stages and already have large digital collections available.

Finnish ethnographer and linguist Matthias Alexander Castrén (1813–1852) produced a large collection of field notes, and also published widely on languages of Northern Eurasia. Recently, two hundred years had passed since his birth, and in this connection the Finno-Ugrian Society launched a project where several of his field notes and grammars are published as commented editions, available both digitally and in print. Numerous monographs have already been published in the series (Salminen et al., 2020; Lehtinen, 2017; Salminen, 2017; Forsberg, 2018; Häkkinen, 2019; Salminen, 2019; Salminen and Janhunen, 2021). The complete series will contain more than twenty volumes. This article discusses the processing of original raw materials up to this point, with a goal of setting a vision of how this process can be refined later on.

Within the research tradition of the Uralic languages, Matthias Castrén is often renowned as the most significant Finnish linguist of the 19th century. Castrén collected vast materials from almost thirty languages on his expeditions to Lapland and Northern Russia between 1838 and 1849 (Janhunen, 2017 updated 2021, 15). The materials are stored in the National Library of Finland. The number of handwritten manuscript pages is approximately ten thousand. Castrén's work carries a unique historical dimension for the languages he studied, and his manuscripts and extensive correspondence with other researchers of the time are also valuable for the history of scientific research.

Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH), pages 47–54
December 19, 2021.

Our study presents individual datasets built from Castrén's materials and reports benchmarks on various text recognition experiments. The main repository for related data is Manuscripta Castreniana collection in Zenodo[1], and other locations are specified when datasets are discussed. We also discuss individual experiments with the text recognition of Castrén's unpublished and published materials, and contextualize the results more widely within early linguistic descriptions. We analyse some of the challenges met in further processing of this content, and delineate possible ways forward. The most important step we can identify is making these materials better available, so that further work can build upon the contributions of more researchers. This is also the step we are trying to help make. We would hope, for example, that eventually Castren's materials would be included in different shared tasks. In the same spirit we also share all our processing code in GitHub[2], which we hope makes these materials easier to access for different researchers in the digital humanities and related fields.

2 Related work

Historical dataset creation is one topic that connects closely to ours. Especially within the Universal Dependencies project (Zeman et al., 2021) there are numerous instances of historical language treebanks. There are five Latin treebanks, Old Church Slavonic treebank (Haug and Jøhndal, 2008), Old Turkish (Derin, 2020) and Old French treebank (Stein and Prévost, 2013), just to mention some of them. Such resources are in a central role, as they allow training NLP models to address different downstream tasks for these language varieties. Naturally, any openly available resource, in plaintext or with annotations, can be used for these purposes. At the same time, the CoNLL-U file format offers a good and well understood structure that can easily be compared.

There are examples of such datasets being used in downstream tasks, such as lemmatizers and POS taggers created for Latin (Clérice, 2021) and Old French (Camps et al., 2021). Work has been done also on Old Swedish (for example, (Borin and Forsberg, 2008; Adesam and Bouma, 2016), but an actual diachronic corpus seems to be still under construction (Pettersson and Borin, 2019). If such resources existed, the analysis of Castrén's 19th century Swedish would be in a different state. There is one unannotated diachronic corpus of Old Literary Finnish (Institute for the Languages of Finland, 2013) and one morpho-syntactically annotated corpus of Mikael Agricola's works (Institute for the Languages of Finland and University of Turku, 2020). The latter has already been used to develop a lemmatizer as well (Hämäläinen et al., 2021). Named entity recognition (NER) for historical publications in Finnish has also received attention lately (Kettunen and Ruokolainen, 2017; Kettunen et al., 2017). A recent survey by Humbel et al. (2021) reviewed different named entity recognition systems for early modern textual documents. Their conclusion was that benchmarking different NER systems in this domain is not currently possible, and suggest wider use of shared forums such as computational linguistics conferences as one way to coordinate further discussion and practices. Study by Idziak et al. (2021) where Polish lexicographic cards were recognized and organized is in some aspects also close to what we would hope to achieve with materials discussed here. To our knowledge, there are no datasets, NLP tools or resources of historical varieties of the endangered languages included in these collections, especially in Castrén's writing system that is essentially an inauguration of a Latin based transcription (Latin transcription with some Cyrillic characters).

3 Materials

We discuss four sections of Castrén's materials. The first consists of ethnographic field notes in 19th century Swedish under the title *Ethnographiska, historiska och statistiska anmärkningar*. Castrén wrote these texts in a extensive area that belongs to the northern regions of the contemporary Russian Federation. This text is also multilingual, with numerous expressions in Cyrillic, but we can approach it largely as a Swedish text. This subset contains 188 pages of handwritten texts. We use this dataset in text recognition experiments reported below, but these materials will be added to the Zenodo collections at a later stage.

The second dataset comes from Tundra Nenets epic poems that have a Russian translation with Swedish commentary. The Figure 1 displays the typical structure in this manuscript. The page is split into two loosely distinguished columns, with Tundra Nenets transcription on the left and the Rus-

[1] https://zenodo.org/communities/castreniana

[2] https://github.com/nikopartanen/manuscripta

sian translation on the right. In the upper right region we see a comment in Swedish in parentheses, but there are also parenthetical clarifications in Russian, as seen in the bottom right corner. All in all, the material comprises 192 pages. This example also provides a good illustration of how the layout detection of these manuscripts is an additional challenge. This dataset is published as is in Zenodo (Castrén, 2021b). The texts have been aligned line by line into the microfilm scans of the original manuscripts in collaboration between the University of Innsbruck and the Finno-Ugrian Society, and this material is an excellent test set for various tasks including text to image alignation, line segmentation and handwritten text recognition.

The third dataset contains published Komi-Zyrian grammar of Castrén (Castrén, 1844) that is written in Latin. The grammar is 174 pages of printed text, all together. In the Manuscripta Castreniana project an English translation with commentary will be published, which adds a new dimension to what kind of computational tasks could be studied with this collection. Additionally, this partially proofread dataset is located in Zenodo (Partanen and Rueter, 2021), with 26 proofread pages of which 3 contain manually constructed tables. Thereby, this dataset is an example of 19th century printed Latin linguistic description, but also serves as the ground truth data for table layout detection as several tables are included with defined table cell structure. This grammar is also available as two different scans, both archived in Zenodo.

The fourth dataset contains Castrén's Komi-Zyrian wedding laments and their transliteration in the modern Komi orthography (Partanen, 2021). These materials were published with Finnish and German translations by Aminoff (1880), and our dataset contains aligned versions of the translations and different transcriptions. Similar dataset could also be created from Castrén's translation of the Gospel of St. Matthew. Crucially, Castrén's transcription system cannot be automatically converted into current orthography as it does not contain all phonemic information that the orthography does. However, the dataset, in itself, is very illustrative of a wider problem in applying NLP to these kinds of materials: the textual representation used has such a different level that, if we cannot transform the transcription into a more modern writing system, we cannot access the text with any current tools.

4 Text recognition

4.1 Background

Text recognition of historical handwritten documents has advanced rapidly in the past few years. The Transkribus platform (Kahle et al., 2017) is leading the field in usability and adoption, and there are reports of consistent results. These include materials by authors such as Foucalt (Massot et al., 2019), Eugène Wilhelm (Schlagdenhauffen, 2020), Jeremy Bentham (Muehlberger et al., 2019, 959) and Konstantin Rychkov (Arkhipov et al., 2021). As mentioned, Castrén's texts include dozens of languages, and Russian, Swedish and Latin are all used as meta languages in different contexts. Similarly presence of different writing systems is also a feature, and challenge, of datasets mentioned above, both with mixed Latin and Greek characters (Schlagdenhauffen, 2020, 4) and Evenki–Russian mixed content (Arkhipov et al., 2021). However, the wide array of endangered languages is still a very specific feature of Castrén's materials.

Currently all text recognition experiments with Castrén's data have been done using the Transkribus platform. The reason for this has been that it allows collaborative editing, and has, at least for handwritten materials, been the currently leading platform. In our further processing the data from Transkribus is exported in Page XML format, which in our experience has been very satisfactory. It appears that Castrén's materials are still particularly challenging to process, and we aim to delineate some of the more technical reasons next.

The first part of the materials was aligned with the microfilm images from XML files where one unaligned transcription version already existed. As these transcriptions were done outside Transkribus, with no visual connection to the actual documents, there may be features in the transcription that should be revised. At the same time the transcriptions were done before the text recognition task was even possible, so the character choices were primarily based on what was convenient for the individual researchers. When tens of thousands of pages are analysed together, it would be important to give careful consideration to which characters should be used to represent which of Castrén's special characters. This work is partially technical and a matter of deciding the correct Unicode characters, but also relates to linguistic analysis. The analysis of the latter type was also conducted for Evenki by Arkhipov and Däbritz (2021).

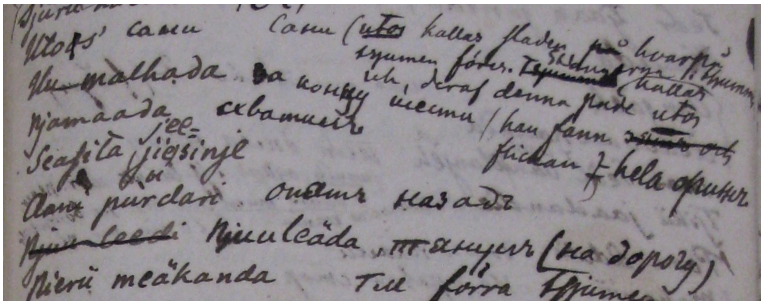

Figure 1: Manuscripta Castreniana, Epic poem 1A, Page 155

As these early versions have been aligned with microfilm scans, and only later have the better quality versions been scanned from the original documents, it may become necessary to realign the transcriptions with these more accurate versions. The materials have been arranged so that such a task is in principle feasible. The second dataset discussed in this study contains exactly these aligned microfilm scans, which, we believe could be used to measure both the impact of chosen character conventions and the quality of scans to the recognition result. The higher quality images are also stored in Zenodo with extensive metadata about the page content (Castrén, 2021a). Generally it is very typical for Castrén's materials that the same text exists in multiple versions. It is unclear to the current authors how to best connect these versions, but we see potentially high value in such an undertaking.

4.2 Experiments

In the current workflow, all texts are manually verified. The ground truth material increases continuously, and has now reached 358 pages. This includes 19,490 lines and approximately 57,000 words. The text recognition accuracy has not significantly improved when the last hundred pages have been added, and the accuracy has been hard to improve further. We first discuss the results with Castrén's printed materials shown in Table 1 and then discuss the handwritten text recognition results shown in Table 2.

Castrén's Komi-Zyrian grammar is written in Latin and it contains individual Komi words and experssions plus some comments in Russian. As Transkribus already contains numerous text recognition models for printed texts, the ideal scenario would be to use some of these directly. We compared some of the Transkribus models for printed texts against the proofread materials, the result being presented in Table 1.

Model	CER %	WER %
Transkribus print 0.3	0.91	4.60
Noscemus GM 5	1.68	8.05
German Kurrent 17th-18th	9.26	38.70
Acta 17 (extended)	10.10	40.23

Table 1: Accuracy on printed Komi-Zyrian grammar written in Latin.

Although Transkribus print 0.3 model[3] does not even include Latin, it still performs extremely well in our test scenario. In the model's documentation CER of 1.6% is reported, and in our experiment the result was even better than that. This has wide significance for work on printed Latin texts, as the out-of-the-box tool truly gives functional result. This should be taken into account when planning further work on printed materials. As expected, the Russian words did not get recognized, and the printed model could benefit from wider inclusion of scripts.

With the handwritten materials the situation is different. We can see in the Table 2 that none of the available HTR models for the Swedish language work very well, even though the result on the Count Records model from the National Archives of Finland is relatively good. As this model is contemporary with Castrén, and also contains handwritten Swedish, the accuracy is not necessarily surprising. Yet, it tells that even with a handwritten text recognition model we do not need to start entirely from scratch.

Even if we were to try to use other models as base models in training, the gains would be relatively minor. Training the Castrén's HTR model with Court Records M10 from the National Archives of Finland as a base model does improve the CER by some percentages compared to Castrén's Ground Truth alone, and on the WER

[3] https://readcoop.eu/model/transkribus-print-multi-language-dutch-german-english-finnish-french-swedish-etc/

Model	CER %	WER %
Castrén (+ NAF base model)	13.19	35.01
Castrén (no base model)	15.40	40.90
NAF Court Records M10	28.65	54.34
Gothenburg Police Reports	32.09	60.50
Edelfelt M13+	34.66	63.59
Stockholm Notaries	43.82	81.23
Jaemtlands domsagas M1+	44.34	78.43

Table 2: Accuracy on Castrén's handwritten Swedish

level the difference is almost five percentage points. We are not seeing entirely transformative differences in the results, but still there is a significant improvement that we get essentially for free.

5 Processing tools

We have archived our processing scripts on GitHub and Zenodo so that they would be maximally useful for a wider community of researchers. Text recognized materials from Transkribus can be exported in Page XML format. The structure is highly standardized, but also relatively complicated. We provide methods to read the lines and their bounding boxes from the XML files into a Python dictionary. After this different operations can be applied, but at a different level: there are already many packages often provide deeper language specific functionality that should be leveraged. Example include UralicNLP (Hämäläinen, 2019) for basic NLP analysis of Uralic languages, and murre for specific dialectal and historical text normalization or lemmatization scenarios (Partanen et al., 2019; Hämäläinen et al., 2021; Hämäläinen et al., 2020). The NLP for Latin also seems fairly developed, and available models could be applied (Clérice, 2021). We see as specific challenges in this the multilinguality and the continuous presence of words and expressions in different languages.

6 Further usage

The materials we have discussed have been created for two purposes: 1) openly licensed ground truth material for text recognition models, and 2) recognized, manually corrected, text for ethnographic and linguistic research. Text recognition models are at the moment line-based and the latter mainly relies on the subject knowledge of the researcher. Neither of these tasks necessarily demands further automatic processing of the materials, at least as long as the research is based on visual use of original versions and the recognized text is used as an aid and search tool to navigate in the document.

However, we consider it still extremely important to be able to extract the text correctly from the files. In the current dataset both the line and layout element structure is indicated by order numbers, and simple concatenation of the lines thereby, in principle, yields the wanted order. However, there are cases where the situation is more complicated. The running order of the elements and lines may not be manually corrected, and it relies on individual conventions whether there is some way to mark whether the order has been verified manually.

In our dataset we find that the running order of the text is generally correct at the page level, and especially so in document pages where layout is simple. This includes the printed Komi-Zyrian grammar and ethnographic notes. In the former table layout detection would need attention, and in the latter problems arise primarily from margin notes and comments between lines. Currently those are not easily placed to the correct locations in the text. In complexly layouted documents with several columns we also find a question of how to indicate the relationship across the columns, as one line is often translation or comment of the other. This issue is seen on almost all pages of the Tundra Nenets epic narrative dataset.

7 Conclusion

Our experiments show that the currently available tools to process 19th century Latin grammar materials in an endangered language can be almost flawlessly recognized with out-of-the-box text recognition models. With handwritten materials the publicly available models need to be customized, but the current accuracy may give at least some starting point if the language and time period match. The divergence of transcription systems and their complex relation to the contemporary orthographies is one challenge that needs to be separately addressed.

To advance actual NLP applications, we also suggest that a sample from Castrén's materials would be published as a treebank or other annotated structure. Such multilingual collection may not fit larger projects such as Universal Dependencies, but similar conventions and file structures could easily be used. How this can be connected to proofread Ground Truth resources, commentaries and digital editions is another question, but there are few materials better for testing this than Castrén's data that is openly available and still acutely relevant for contemporary research.

References

Yvonne Adesam and Gerlof Bouma. 2016. Old Swedish part-of-speech tagging between variation and external knowledge. In *Proceedings of the 10th SIGHUM workshop on language technology for cultural heritage, social sciences, and humanities*, pages 32–42.

T. G. Aminoff. 1880. Syrjäniläisiä häälauluja, koonnut M. A. Castrén, alkutekstistä suomentanut ja saksalaisella käännöksellä varustanut T. G. Aminoff. *Acta Societatis Scientiarum Fennicae*, XI:203–231.

Alexandre Arkhipov, Anna Barinskaya, and Roman Shtefura. 2021. Using handwritten text recognition on bilingual Evenki-Russian manuscripts of Konstantin Rychkov. *Scripta & E-Scripta*, 21.

A.V. Arkhipov and C.L. Däbritz. 2021. Reconstructing phonetics behind the graphic system of Evenki texts from the Rychkov archive. *Rhema*, (2):46–64.

Lars Borin and Markus Forsberg. 2008. Something old, something new: A computational morphological description of Old Swedish. In *LREC 2008 workshop on language technology for cultural heritage data (LaTeCH 2008)*, pages 9–16.

Jean-Baptiste Camps, Thibault Clérice, Frédéric Duval, Naomi Kanaoka, Ariane Pinche, et al. 2021. Corpus and models for Lemmatisation and POS-tagging of Old French. *arXiv preprint arXiv:2109.11442*.

Matthias Alexander Castrén. 1844. *Elementa grammatices Syrjaenae conscripsit MA Castrén*. Ex officina typographica heredum Simelii.

M. A. Castrén. 2021a. Mc viii samoiedica 2: Jurak-samoiedica 1.

M. A. Castrén. 2021b. MC VIII SAMOIEDICA 2: JURAK-SAMOIEDICA 1: Line-aligned Ground Truth.

Thibault Clérice. 2021. Latin Lasla Model. Zenodo, 10.5281/zenodo.4661034.

Mehmet Oguz Derin. 2020. Ud_old_turkish-tonqq. Https://github.com/UniversalDependencies/UD_Old_Turkish-Tonqq.

Ulla-Maija Forsberg, editor. 2018. *Ostiacica*, volume Linguistica V of *Manuscripta Castreniana*. Finno-Ugrian Society.

Mika Hämäläinen, Khalid Alnajjar, Niko Partanen, and Jack Rueter. 2021. Finnish dialect identification: The effect of audio and text. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8777–8783, United States. The Association for Computational Linguistics.

Mika Hämäläinen, Niko Partanen, and Khalid Alnajjar. 2020. Normalization of different Swedish dialects spoken in Finland. In *GeoHumanities'20: Proceedings of the 4th ACM SIGSPATIAL Workshop on Geospatial Humanities*, page 24–27, United States. ACM.

Dag TT Haug and Marius Jøhndal. 2008. Creating a parallel treebank of the old Indo-European Bible translations. In *Proceedings of the second workshop on language technology for cultural heritage data (LaTeCH 2008)*, pages 27–34.

Marco Humbel, Julianne Nyhan, Andreas Vlachidis, Kim Sloan, and Alexandra Ortolja-Baird. 2021. Named-entity recognition for early modern textual documents: a review of capabilities and challenges with strategies for the future. *Journal of Documentation*.

Kaisa Häkkinen, editor. 2019. *Fennica*, volume Linguistica I of *Manuscripta Castreniana*. Finno-Ugrian Society.

Mika Hämäläinen. 2019. UralicNLP: An NLP library for Uralic languages. *Journal of Open Source Software*, 4(37):1345.

Mika Hämäläinen, Niko Partanen, and Khalid Alnajjar. 2021. Lemmatization of historical old literary Finnish texts in modern orthography. In *Actes de la Conférence sur le Traitement Automatique des Langues Naturelles (TALN)*.

Jan Idziak, Artjoms Šeļa, Michał Woźniak, Albert Leśniak, Joanna Byszuk, and Maciej Eder. 2021. Scalable handwritten text recognition system for lexicographic sources of under-resourced languages and alphabets. In *International Conference on Computational Science*, pages 137–150. Springer.

Institute for the Languages of Finland. 2013. Corpus of Old Literary Finnish.

Institute for the Languages of Finland and University of Turku. 2020. The Morpho-Syntactic Database of Mikael Agricola's Works version 1.1.

Juha Janhunen. 2017 updated 2021. *Manuscripta Castreniana: A General Preface to the Series*. Manuscripta Castreniana. Finno-Ugrian Society.

Philip Kahle, Sebastian Colutto, Günter Hackl, and Günter Mühlberger. 2017. Transkribus-a service platform for transcription, recognition and retrieval of historical documents. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 4, pages 19–24. IEEE.

Kimmo Kettunen, Eetu Mäkelä, Teemu Ruokolainen, Juha Kuokkala, and Laura Löfberg. 2017. Old content and modern tools: Searching named entities in a Finnish OCRed historical newspaper collection 1771–1910. *Digital Humanities Quarterly*, 11(3).

Kimmo Kettunen and Teemu Ruokolainen. 2017. Names, right or wrong: Named entities in an OCRed historical Finnish newspaper collection. In *Proceedings of the 2nd International Conference on Digital Access to Textual Cultural Heritage*, pages 181–186.

Ildikó Lehtinen, editor. 2017. *Collectiones museorum*, volume Realia II, Ethnographica 1 of *Manuscripta Castreniana*. Finno-Ugrian Society.

Marie-Laure Massot, Arianna Sforzini, and Vincent Ventresque. 2019. Transcribing Foucault's handwriting with Transkribus. *Journal of Data Mining and Digital Humanities*.

Guenter Muehlberger, Louise Seaward, Melissa Terras, Sofia Ares Oliveira, Vicente Bosch, Maximilian Bryan, Sebastian Colutto, Hervé Déjean, Markus Diem, Stefan Fiel, et al. 2019. Transforming scholarship in the archives through handwritten text recognition: Transkribus as a case study. *Journal of documentation*, 75(5):954–976.

Niko Partanen. 2021. nikopartanen/castren-komi-wedding-laments: Matthias Alexander Castrén's Komi Wedding Laments, sentence-aligned dataset.

Niko Partanen, Khalid Alnajjar, Mika Hämäläinen, and Jack Rueter. 2021. Linguistic change and historical periodization of Old Literary Finnish. In *Proceedings of the 2nd International Workshop on Computational Approaches to Historical Language Change 2021*, page 21–27, United States. The Association for Computational Linguistics.

Niko Partanen, Mika Hämäläinen, and Khalid Alnajjar. 2019. Dialect text normalization to normative standard finnish. In *The Fifth Workshop on Noisy User-generated Text (W-NUT 2019)*, page 141–146, United States. The Association for Computational Linguistics.

Niko Partanen and Jack Rueter. 2021. Castren 1844: Elementa grammaticae Syrjaenae, OCR Ground Truth.

Eva Pettersson and Lars Borin. 2019. Towards a Swedish diachronic corpus: Intended content, structure and format of version 1.0. In *SWE-CLARIN REPORT SERIES*. SWE-CLARIN.

Tanja Säily, Eetu Mäkelä, and Mika Hämäläinen. 2021. From plenipotentiary to puddingless: Users and uses of new words in early English letters. In *Multilingual Facilitation*, pages 153–169, Finland. University of Helsinki.

Tapani Salminen, Karina Lukin, and Petri-Tapio Heikkonen, editors. 2020. *Jurak-Samoiedica*. Manuscripta Castreniana. Finno-Ugrian Society.

Timo Salminen, editor. 2017. *Archaeologica et historica; Universitaria*, volume Realia I of *Manuscripta Castreniana*. Finno-Ugrian Society.

Timo Salminen, editor. 2019. *Itineraria (1–2)*, volume Personalia II, 1–2 of *Manuscripta Castreniana*. Finno-Ugrian Society.

Timo Salminen and Juha Janhunen, editors. 2021. *Epistulae*, volume Personalia I of *Manuscripta Castreniana*. Finno-Ugrian Society.

Régis Schlagdenhauffen. 2020. Optical recognition assisted transcription with Transkribus: The experiment concerning Eugène Wilhelm's personal diary (1885-1951). *Journal of Data Mining and Digital Humanities*, 335.

Achim Stein and Sophie Prévost. 2013. Syntactic annotation of medieval texts: the Syntactic Reference Corpus of Medieval French (SRCMF). In P. Bennett, M. Durrell, S. Scheible, and R. Whitt, editors, *New Methods in Historical Corpus Linguistics*, Corpus Linguistics and International Perspectives on Language, pages 275–282. Narr Verlag.

Daniel Zeman, Joakim Nivre, Mitchell Abrams, Elia Ackermann, Noëmi Aepli, Hamid Aghaei, Željko Agić, Amir Ahmadi, Lars Ahrenberg, Chika Kennedy Ajede, Gabrielė Aleksandravičiūtė, Ika Alfina, Lene Antonsen, Katya Aplonova, Angelina Aquino, Carolina Aragon, Maria Jesus Aranzabe, Bilge Nas Arıcan, Hórunn Arnardóttir, Gashaw Arutie, Jessica Naraiswari Arwidarasti, Masayuki Asahara, Deniz Baran Aslan, Luma Ateyah, Furkan Atmaca, Mohammed Attia, Aitziber Atutxa, Liesbeth Augustinus, Elena Badmaeva, Keerthana Balasubramani, Miguel Ballesteros, Esha Banerjee, Sebastian Bank, Verginica Barbu Mititelu, Starkaur Barkarson, Victoria Basmov, Colin Batchelor, John Bauer, Seyyit Talha Bedir, Kepa Bengoetxea, Gözde Berk, Yevgeni Berzak, Irshad Ahmad Bhat, Riyaz Ahmad Bhat, Erica Biagetti, Eckhard Bick, Agnė Bielinskienė, Kristín Bjarnadóttir, Rogier Blokland, Victoria Bobicev, Loïc Boizou, Emanuel Borges Völker, Carl Börstell, Cristina Bosco, Gosse Bouma, Sam Bowman, Adriane Boyd, Anouck Braggaar, Kristina Brokaitė, Aljoscha Burchardt, Marie Candito, Bernard Caron, Gauthier Caron, Lauren Cassidy, Tatiana Cavalcanti, Gülşen Cebiroğlu Eryiğit, Flavio Massimiliano Cecchini, Giuseppe G. A. Celano, Slavomír Čéplö, Neslihan Cesur, Savas Cetin, Özlem Çetinoğlu, Fabricio Chalub, Shweta Chauhan, Ethan Chi, Taishi Chika, Yongseok Cho, Jinho Choi, Jayeol Chun, Alessandra T. Cignarella, Silvie Cinková, Aurélie Collomb, Çağrı Çöltekin, Miriam Connor, Marine Courtin, Mihaela Cristescu, Philemon. Daniel, Elizabeth Davidson, Marie-Catherine de Marneffe, Valeria de Paiva, Mehmet Oguz Derin, Elvis de Souza, Arantza Diaz de Ilarraza, Carly Dickerson, Arawinda Dinakaramani, Elisa Di Nuovo, Bamba Dione, Peter Dirix, Kaja Dobrovoljc, Timothy Dozat, Kira Droganova, Puneet Dwivedi, Hanne Eckhoff, Sandra Eiche, Marhaba Eli, Ali Elkahky, Binyam Ephrem, Olga Erina, Tomaž Erjavec, Aline Etienne, Wograine Evelyn, Sidney Facundes, Richárd Farkas, Marília Fernanda, Hector

Fernandez Alcalde, Jennifer Foster, Cláudia Freitas, Kazunori Fujita, Katarína Gajdošová, Daniel Galbraith, Marcos Garcia, Moa Gärdenfors, Sebastian Garza, Fabrício Ferraz Gerardi, Kim Gerdes, Filip Ginter, Gustavo Godoy, Iakes Goenaga, Koldo Gojenola, Memduh Gökırmak, Yoav Goldberg, Xavier Gómez Guinovart, Berta González Saavedra, Bernadeta Griciūtė, Matias Grioni, Loïc Grobol, Normunds Grūzītis, Bruno Guillaume, Céline Guillot-Barbance, Tunga Güngör, Nizar Habash, Hinrik Hafsteinsson, Jan Hajič, Jan Hajič jr., Mika Hämäläinen, Linh Hà Mỹ, Na-Rae Han, Muhammad Yudistira Hanifmuti, Sam Hardwick, Kim Harris, Dag Haug, Johannes Heinecke, Oliver Hellwig, Felix Hennig, Barbora Hladká, Jaroslava Hlaváčová, Florinel Hociung, Petter Hohle, Eva Huber, Jena Hwang, Takumi Ikeda, Anton Karl Ingason, Radu Ion, Elena Irimia, Ọlájídé Ishola, Kaoru Ito, Tomáš Jelínek, Apoorva Jha, Anders Johannsen, Hildur Jónsdóttir, Fredrik Jørgensen, Markus Juutinen, Sarveswaran K, Hüner Kaşıkara, Andre Kaasen, Nadezhda Kabaeva, Sylvain Kahane, Hiroshi Kanayama, Jenna Kanerva, Neslihan Kara, Boris Katz, Tolga Kayadelen, Jessica Kenney, Václava Kettnerová, Jesse Kirchner, Elena Klementieva, Arne Köhn, Abdullatif Köksal, Kamil Kopacewicz, Timo Korkiakangas, Natalia Kotsyba, Jolanta Kovalevskaitė, Simon Krek, Parameswari Krishnamurthy, Oğuzhan Kuyrukçu, Aslı Kuzgun, Sookyoung Kwak, Veronika Laippala, Lucia Lam, Lorenzo Lambertino, Tatiana Lando, Septina Dian Larasati, Alexei Lavrentiev, John Lee, Phng Lê H`ông, Alessandro Lenci, Saran Lertpradit, Herman Leung, Maria Levina, Cheuk Ying Li, Josie Li, Keying Li, Yuan Li, KyungTae Lim, Bruna Lima Padovani, Krister Lindén, Nikola Ljubešić, Olga Loginova, Andry Luthfi, Mikko Luukko, Olga Lyashevskaya, Teresa Lynn, Vivien Macketanz, Aibek Makazhanov, Michael Mandl, Christopher Manning, Ruli Manurung, Büşra Marşan, Cătălina Mărănduc, David Mareček, Katrin Marheinecke, Héctor Martínez Alonso, André Martins, Jan Mašek, Hiroshi Matsuda, Yuji Matsumoto, Alessandro Mazzei, Ryan McDonald, Sarah McGuinness, Gustavo Mendonça, Niko Miekka, Karina Mischenkova, Margarita Misirpashayeva, Anna Missilä, Cătălin Mititelu, Maria Mitrofan, Yusuke Miyao, AmirHossein Mojiri Foroushani, Judit Molnár, Amirsaeid Moloodi, Simonetta Montemagni, Amir More, Laura Moreno Romero, Giovanni Moretti, Keiko Sophie Mori, Shinsuke Mori, Tomohiko Morioka, Shigeki Moro, Bjartur Mortensen, Bohdan Moskalevskyi, Kadri Muischnek, Robert Munro, Yugo Murawaki, Kaili Müürisep, Pinkey Nainwani, Mariam Nakhlé, Juan Ignacio Navarro Horñiacek, Anna Nedoluzhko, Gunta Nešpore-Bērzkalne, Manuela Nevaci, Lng Nguy˜ên Thị, Huy`ên Nguy˜ên Thị Minh, Yoshihiro Nikaido, Vitaly Nikolaev, Rattima Nitisaroj, Alireza Nourian, Hanna Nurmi, Stina Ojala, Atul Kr. Ojha, Adédayọ̀ Olúòkun, Mai Omura, Emeka Onwuegbuzia, Petya Osenova, Robert Östling, Lilja Øvrelid, Şaziye Betül Özateş, Merve Özçelik, Arzucan Özgür, Balkız Öztürk Başaran, Hyunji Hayley Park, Niko Partanen, Elena Pascual, Marco Passarotti, Agnieszka Patejuk, Guilherme Paulino-Passos, Angelika Peljak-Łapińska, Siyao Peng, Cenel-Augusto Perez, Natalia Perkova, Guy Perrier, Slav Petrov, Daria Petrova, Jason Phelan, Jussi Piitulainen, Tommi A Pirinen, Emily Pitler, Barbara Plank, Thierry Poibeau, Larisa Ponomareva, Martin Popel, Lauma Pretkalniņa, Sophie Prévost, Prokopis Prokopidis, Adam Przepiórkowski, Tiina Puolakainen, Sampo Pyysalo, Peng Qi, Andriela Rääbis, Alexandre Rademaker, Taraka Rama, Loganathan Ramasamy, Carlos Ramisch, Fam Rashel, Mohammad Sadegh Rasooli, Vinit Ravishankar, Livy Real, Petru Rebeja, Siva Reddy, Georg Rehm, Ivan Riabov, Michael Rießler, Erika Rimkutė, Larissa Rinaldi, Laura Rituma, Luisa Rocha, Eiríkur Rögnvaldsson, Mykhailo Romanenko, Rudolf Rosa, Valentin Roșca, Davide Rovati, Olga Rudina, Jack Rueter, Kristján Rúnarsson, Shoval Sadde, Pegah Safari, Benoît Sagot, Aleksi Sahala, Shadi Saleh, Alessio Salomoni, Tanja Samardžić, Stephanie Samson, Manuela Sanguinetti, Ezgi Sanıyar, Dage Särg, Baiba Saulīte, Yanin Sawanakunanon, Shefali Saxena, Kevin Scannell, Salvatore Scarlata, Nathan Schneider, Sebastian Schuster, Lane Schwartz, Djamé Seddah, Wolfgang Seeker, Mojgan Seraji, Mo Shen, Atsuko Shimada, Hiroyuki Shirasu, Yana Shishkina, Muh Shohibussirri, Dmitry Sichinava, Janine Siewert, Einar Freyr Sigursson, Aline Silveira, Natalia Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Mária Šimková, Kiril Simov, Maria Skachedubova, Aaron Smith, Isabela Soares-Bastos, Carolyn Spadine, Rachele Sprugnoli, Steinþór Steingrímsson, Antonio Stella, Milan Straka, Emmett Strickland, Jana Strnadová, Alane Suhr, Yogi Lesmana Sulestio, Umut Sulubacak, Shingo Suzuki, Zsolt Szántó, Dima Taji, Yuta Takahashi, Fabio Tamburini, Mary Ann C. Tan, Takaaki Tanaka, Samson Tella, Isabelle Tellier, Marinella Testori, Guillaume Thomas, Liisi Torga, Marsida Toska, Trond Trosterud, Anna Trukhina, Reut Tsarfaty, Utku Türk, Francis Tyers, Sumire Uematsu, Roman Untilov, Zdeňka Urešová, Larraitz Uria, Hans Uszkoreit, Andrius Utka, Sowmya Vajjala, Rob van der Goot, Martine Vanhove, Daniel van Niekerk, Gertjan van Noord, Viktor Varga, Eric Villemonte de la Clergerie, Veronika Vincze, Natalia Vlasova, Aya Wakasa, Joel C. Wallenberg, Lars Wallin, Abigail Walsh, Jing Xian Wang, Jonathan North Washington, Maximilan Wendt, Paul Widmer, Seyi Williams, Mats Wirén, Christian Wittern, Tsegay Woldemariam, Tak-sum Wong, Alina Wróblewska, Mary Yako, Kayo Yamashita, Naoki Yamazaki, Chunxiao Yan, Koichi Yasuoka, Marat M. Yavrumyan, Arife Betül Yenice, Olcay Taner Yıldız, Zhuoran Yu, Zdeněk Žabokrtský, Shorouq Zahra, Amir Zeldes, Hanzhi Zhu, Anna Zhuravleva, and Rayan Ziane. 2021. Universal Dependencies 2.8.1. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Lotte and Annette: A Framework for Finding and Exploring Key Passages in Literary Works

Frederik Arnold and **Robert Jäschke**
Humboldt-Universität zu Berlin
{frederik.arnold, robert.jaeschke}@hu-berlin.de

Abstract

We present an approach that leverages expert knowledge contained in scholarly works to automatically identify key passages in literary works. Specifically, we extend a text reuse detection method for finding quotations, such that our system *Lotte* can deal with common properties of quotations, for example, ellipses or inaccurate quotations. An evaluation shows that Lotte outperforms four existing approaches. To generate key passages, we combine overlapping quotations from multiple scholarly texts. An interactive website, called *Annette*, for visualizing and exploring key passages makes the results accessible and explorable.

1 Introduction

Identification of key passages in nonfiction has long been a topic of research. For example, in the context of text summarization to identify sentences and passages which contain key arguments (Paice, 1980). While there has been a lot of progress for nonfiction (Yao et al., 2017), there are no working solutions for fiction.

In this paper, we present a first step towards a system to automatically identify key passages in fiction. We understand *key passages* as passages that are particularly important to expert readers, following a general definition of "key words" (Scott and Tribble, 2006). We leverage the expert knowledge contained in scholarly works to automatically identify potential key passages. Authors of scholarly works use different types of citations to refer to original works, for example, quotations or paraphrases. We adapt existing methods for text re-use detection (Grune and Huntjens, 1989) such that our system can deal with common properties of quotations such as ellipses or unclean quotations, for instance, missing words or spelling mistakes. On top of that, our system works independently of the order of quotations and can handle multiple quotations of the same text. To generate key passages, we combine overlapping quotations from multiple scholarly works.

Our contributions are *Lotte*, an algorithm and Python tool for quotation detection in fictional texts and *Annette*, an interactive website for visualizing and exploring key passages. This makes key passages available at a larger scale and in structured form and opens up many new opportunities for analyses in literary studies and the praxeology of literary studies.

This paper is organized as follows: In Section 2, we provide an overview on related work. In Section 3, we present our approach to finding key passages. In Section 4, we describe our evaluation setup including four existing systems and in Section 5 we show how our approach outperforms them. Finally, in Section 6, we present our tool for visualizing and exploring key passages.

2 Related Work

Quotation detection can be regarded as a kind of text reuse detection, which is frequently applied for plagiarism detection (Hoad and Zobel, 2003). There, the goal is to find quotations and citations without proper attribution. In our case, we assume proper attribution and focus on the step of *finding* and *linking* quotations.

Several tools for different use cases try to solve similar or related problems. For example, *BLAST* aligns biological sequences (Altschul et al., 1990) and has been adopted for text reuse detection (Vesanto et al., 2017a,b). Copyfind (Bloomfield) is an open source tool for comparing documents written in C++. While *Passim* (Smith et al., 2014) and *TextMatcher* (Reeve, 2020) are simple text reuse detection tools, *TRACER* (Büchler, 2016) is an elaborate framework consisting of around 700

Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH), pages 55–63
December 19, 2021.

algorithms. *SIM* (Grune and Huntjens, 1989) finds lexical similarities in source code and natural language texts, originally built to find duplicate code in large code bases. The original idea worked well enough to be used to find copied work in student submissions. SIM works with a number of programming languages and can easily be extended to work with new languages by providing a lexical description. *Sim_text* is a version of SIM for checking duplicates in natural language texts. Based on SIM, *similarity texter (SimT)* is a tool for text comparison written in JavaScript by Kalaidopoulou (2016).

For various reasons, these tools are not appropriate for detecting key passages. For example, TextMatcher only finds quotations that appear in the same order in both texts. None of the tools can find multiple quotations of the same text. In Section 5 we evaluate BLAST, Copyfind, SimT, and TextMatcher and compare them against our approach. We did not evaluate TRACER, as we could not manage to extract exact matches. Passim is the only system we could not get to work at all as its dependencies were no longer available.

A website for visualizing (literal) citations of Shakespeare's works has been presented by Miller. It visualizes how often each line from every play has been cited in JSTOR's journal collection. The website is limited to the visualization of the citation frequency of each line and does not offer any functionality to explore the source of citations.

3 Lotte – A Text Reuse Detection Tool

In this section, we describe our approach for identifying quotations which solves the following task:[1] Given a *source* and a *target* text, it finds all instances where the target text contains some part of the source text.

Our approach is based on a modified and extended version of Sim_text by Grune and Huntjens (1989). The original implementation is written in C while our reimplementation is in Python. Reimplementing the algorithm allowed us to integrate extensions for properly handling specific properties of quotations which are not covered by Sim_text.

The algorithm works in five main steps which we describe in the sequel. Table 1 shows two simplified example texts which consist of words only without any punctuation except for periods and one ellipsis in the target text. The numbers on the left are the positions of the words (w_1 in the source text is at position 0, w_2 at position 1, w_8 at position 7, etc.).

Source text
^{0} w_1 w_2 w_3 w_4 w_5 w_6 w_7 w_8. w_4
^{9} w_5 w_6 w_7 w_8 w_9. w_{10} w_{11} w_{12}
$^{17}w_{13}$ w_{14} w_{15} w_{16} w_{17} w_{18} w_{19}
$^{24}w_{20}$ w_{21} w_{22} w_{23} w_{24} w_{25} w_{26}
Target text
$^{50}w_1$ w_2 w_3 w_{11} w_{12} w_{13} [...] w_{23} w_{24}
$^{59}w_{25}$ w_{26}. w_{30} w_{31} w_{32} w_{33} w_{34} w_4
$^{67}w_5$ w_6 w_7 w_8 w_9. w_{10} w_{11} w_4 w_5 w_6

Table 1: Example source and target texts.

Word sequences	Starting positions
w_1 w_2 w_3	[0]
w_2 w_3 w_4	[1]
w_4 w_5 w_6	[3, 8]

Table 2: Some word sequences with starting positions of the source text.

3.1 Step 1: Tokenize Text

Both texts are cleaned and tokenized and sequences like '...', '[...]' and other possible variants of ellipses-indicating characters are masked so they can later be identified easily. Punctuation indicating the end of a sentence is also masked for the same reason. All other special characters and numbers are removed. Finally, the text is tokenized using white space characters. The main improvement to Sim_text is the masking of characters which carry information needed later.

3.2 Step 2: Initial Positions

A mapping of word sequences to starting positions for the source text is created. The initial sequence length is currently hard-coded to three as it worked best in our tests. The value can easily be changed but a smaller value results in too many initial matches which will be removed later anyway because of the minimal length cutoff for results. Table 2 shows examples for word sequences and their starting positions. The sequence w_1 w_2 w_3 starts at position 0, sequence w_2 w_3 w_4 at position 1, etc. The same sequence can appear multiple times and

[1] The source code is licensed under the Apache License 2.0 and available at https://scm.cms.hu-berlin.de/schluesselstellen/lotte.

Source text	Target text
0	[0]
3	[67]
8	[67]

Table 3: Some source text starting positions with corresponding target text starting positions.

therefore might have multiple starting positions. This handling of sequences that appear multiple times is the main improvement to Sim_text.

3.3 Step 3: Forward References

A table of forward references, that is, a mapping of starting positions in the source text to a list of starting positions in the target text is created. For example, the sequence $w_1\ w_2\ w_3$ which starts at position 0 in the source text can also be found in the target text starting at position 0.

As Sim_text only considers exact matches, we improved this to use MinHash Locality Sensitive Hashing (Slaney and Casey, 2008) and Levenshtein distance (Levenshtein, 1966) to find the best matching sequence. Our algorithm first gets a list of all possible matches above a similarity threshold of 0.95. From that list, it then selects the best match with a normalized Levenshtein distance[2] equal or greater 0.9. These thresholds were optimized using expert knowledge (cf. Section 4.2).

3.4 Step 4: Extend Initial Matches

The initially three token long matches are extended forwards and backwards to match longer sequences, if possible. For example, in Table 1 the initially matched sequence $w_4\ w_5\ w_6$ can be extended forward to match $w_4\ w_5\ w_6\ w_7\ w_8\ w_9\ w_{10}\ w_{11}$. Backwards extension is needed to handle certain edge cases occurring due to ellipses or mismatches of tokens. Sim_text does neither include backwards extension nor handling of ellipses or mismatches of tokens. Our algorithm also uses the normalized Levenshtein distance for token matching.

3.5 Step 5: Reprocess Found Matches

Step 4 extends the initial matches in a relatively conservative manner. A more aggressive approach would lead to too many false positives. This means that the quality of the matches can be further increased by reprocessing the intermediate matches. This is implemented in the following novel steps.

[2] https://github.com/maxbachmann/RapidFuzz

Start	End	Match segments
50	61	$w_{11}\ w_{12}\ w_{13}$
12	23	$w_{11}\ w_{12}\ w_{13}$
98	113	$w_{23}\ w_{24}\ w_{25}\ w_{26}.$
28	44	$w_{23}\ w_{24}\ w_{25}\ w_{26}.$
9	25	$w_4\ w_5\ w_6\ w_7\ w_8.$
65	79	$w_4\ w_5\ w_6\ w_7\ w_8$
26	53	$w_4\ w_5\ w_6\ w_7\ w_8\ w_9.\ w_{10}\ w_{11}$
65	91	$w_4\ w_5\ w_6\ w_7\ w_8\ w_9.\ w_{10}\ w_{11}$

Table 4: Intermediate results after Step 4. First line of each pair (red) is the match segment from source text and the second line (blue) is the match segment from the target text.

Table 4 shows some of the intermediate results after executing Steps 1 to 4. The first line of a pair is the match segment from the source text and the second line is the match segment from the target text. The two numbers at the beginning of a line correspond to the first and last character's position of a match in the original text, respectively.

Neighbouring Matches The first improvement is to merge neighbouring matches. The intermediate matches are sorted in order of appearance in the source text. They are then checked for matches that appear neighbouring in the target text and in the source text and are not further apart than a certain number of tokens. If there is an ellipsis between the two matches in the target text, the number of tokens between the matches can be greater.

Overlapping Segments We remove matches with overlapping target match segments. In Table 4, the last two matches completely overlap in the target text. This means that one of the matches has to be removed. In such a case only the longer one will be kept.

Short Matches The remaining matches are checked for matches which are shorter than a certain length, which can be defined by the user. In our case, we only keep matches which are five words or longer. All other matches are removed.

Sentence Boundaries Finally, we check for matches that cross sentence boundaries. This happens in a number of cases where after a match the

Start	End	Match segments
50	113	w_{11} w_{12} w_{13} w_{14} w_{15} w_{16} w_{17} w_{18} w_{19} w_{20} w_{21} w_{22} w_{23} w_{24} w_{25} w_{26}
12	44	w_{11} w_{12} w_{13} [...] w_{23} w_{24} w_{25} w_{26}.
26	44	w_4 w_5 w_6 w_7 w_8 w_9.
65	83	w_4 w_5 w_6 w_7 w_8 w_9.

Table 5: Final results. For both matches, the first part (red) is from the source text and the second part (blue) is the match segment from the target text.

source and target text continue with the same words by chance. We check for matches which end with a sentence delimiter (., ;, ! and ?) followed by one or two words. In such cases the words after the delimiter are removed. The final results are shown in Table 5.

4 Experiments

4.1 Datasets

We evaluate our approach on two literary works, *Die Judenbuche* by Annette von Droste-Hülshoff (1979) and *Michael Kohlhaas* by Heinrich von Kleist (1978), with 44 and 49 interpretive scholarly articles, respectively.[3] The texts were annotated in the ArguLIT project (Winko, 2017–2020) using TEI/XML (TEI Consortium, eds.). The corpus contains annotations for quotations of different types, for example, quotations from the primary literary work, other literary works, or other scholarly works. Only clearly marked quotations, that is, with quotation marks, were annotated. For the purposes of this evaluation, we are only interested in *quotations from the primary literary work*. Table 6 shows the number of articles and quotations from the primary literary work with a length of five or more words ("gold items").

We limit the experiments to finding matches of five or more words because none of the approaches works for very short matches. It would be possible to find shorter matches but it introduces too much noise. The limit is based on the distribution of all word *n*-grams which have a frequency of at least two (cf. Table 7). The counts are calculated after removing special characters and only the longest sequence is counted, for example, for a 7-gram, the 3- and 4-sub-grams are not counted again. *Die Judenbuche* contains two 5-grams, one 6-gram, and one 7-gram which appear twice. This is few enough to not introduce too much noise. In the case of *Michael Kohlhaas*, which is twice as long as *Die Judenbuche*, the *n*-gram counts do not as clearly support a limit of five or more words. We decided to keep the limit but this could be improved in the future. For example, as a first step, Lotte could report ambiguous cases. In the longer term, we will develop methods for extracting quotations shorter than five words and handling ambiguous cases.

Literary work	Die Judenbuche	Michael Kohlhaas
Scholarly articles	44	49
Gold items (≥ 5 words)	1 235	1 349
Quotations with ellipses	206	262
Literary text characters	102 477	221 097
Scholarly articles characters	2 650 095	2 778 528

Table 6: Basic statistics for *Die Judenbuche* and *Michael Kohlhaas*.

n-gram	Frequency 2		3		4		5		> 5	
	J	K	J	K	J	K	J	K	J	K
3	130	752	14	114	3	27	1	4	2	7
4	21	176	1	21	0	3	0	2	0	0
5	2	55	0	7	0	1	0	0	0	0
6	1	11	0	1	0	0	0	0	0	1
7	1	3	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	0	1	0	0	0	0	0	0	0	0
10	0	2	0	0	0	0	0	0	0	0

Table 7: The *n*-gram counts for *Die Judenbuche* and *Michael Kohlhaas*.

4.2 Setup

For each approach, we try to select parameters as close as possible to those of our approach. Minimal match length is always set to 5. Lotte's thresholds and parameters were optimized on the corpus for *Die Judenbuche*. The results will show that the approach performs equally well on unseen texts.

BLAST There are several parameters but none really correspond to those of the other approaches.

[3] For the sake of brevity, we will reference *Die Judenbuche* and *Michael Kohlhaas* with J and K.

	BLAST	Copyfind	Lotte	SimT	TextMatcher
Order independent	✓	✓	✓	✓	-
one-to-many matching	-	-	✓	-	-
Fuzzy matching	✓	-	✓	-	-
Skip words	-	✓	✓	-	✓
Ellipsis handling	-	-	✓	-	-

Table 8: System functionality comparison.

So we use the defaults and remove short matches in a post-processing step. BLAST requires a mapping from characters to DNA sequence blocks. Using the provided mapping for English with space worked better than using a mapping based on the most frequent characters in German.

Copyfind We ignore letter case, numbers and punctuation. We allow up to two non-matching words between perfectly matching phrases and a minimum of 80 % matching words for a phrase to be considered a match.

Lotte We use the following parameters: A look-back limit of 10, a look-ahead limit of 3, a maximum merge distance of 2, and a maximum merge distance for ellipses of 10. We ignore letter case, numbers, punctuation, and replace umlauts.

SimT We ignore letter case, numbers and punctuation and replace umlauts.

TextMatcher Again, there are several parameters but none really correspond to those of the other approaches. We set threshold and cutoff to 0 and leave the default value of 3 for n-gram size. We also remove short matches in a post-processing step.

Table 8 shows a comparison of the functionality of each approach. TextMatcher is the only system that does not support order-independent matching, that is, only matches appearing in the same order in both texts will be found. One-to-many matching, that is, matching a sequence in the source text with multiple sequences in the target text is only supported by Lotte. Fuzzy matching is supported by Lotte and BLAST. Copyfind, Lotte, and TextMatcher can skip words, that is, a sequence can still be a match even if there is a mismatch between individual words. Lotte is the only system that explicitly handles ellipses. Processing 44 scholarly works for *Die Judenbuche* with Lotte takes around five minutes on an Intel Core i9-9880H CPU.

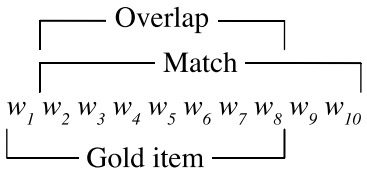

Figure 1: Calculation of precision (|Overlap| / |Match|), recall (|Overlap| / |Gold item|), and F_1-score based on the overlap between a match and a gold item.

4.3 Evaluation

For the evaluation, we asses the performance of all approaches by averaging precision, recall, and F_1-score of each match and gold item. Figure 1 illustrates the calculation. Internally, we use character counts for the calculation. This ensures that the results of all approaches are comparable and is necessary in case an approach does not respect token boundaries and returns incomplete words. Matches which cover multiple gold items are punished by taking the average precision. Analogously, gold items which are partly covered by multiple matches are punished by taking the average recall.

5 Results

5.1 Performance Comparison

Table 9 shows the performance of the approaches in the top section. The bottom section shows different variants of Lotte which we discuss in Section 5.3.

For *Die Judenbuche*, Lotte outperforms the other approaches with an F_1-score of 0.86. Copyfind performs second best (0.79), closely followed by SimT (0.76). SimT's precision is highest with 0.91.

For *Michael Kohlhaas* the results look different. Lotte achieves the highest recall of 0.90, but SimT performs best with the highest precision of 0.83 and an F_1-score of 0.79.

5.2 Error Analysis

To better understand the differences in precision between the approaches and the lower precision

Approach	Die Judenbuche			Michael Kohlhaas		
	Precision	Recall	F_1	Precision	Recall	F_1
BLAST	0.59	0.61	0.60	0.37	0.59	0.45
Copyfind	0.85	0.75	0.79	0.76	0.79	0.78
SimT	**0.91**	0.64	0.76	**0.83**	0.74	**0.79**
TextMatcher	0.69	0.37	0.48	0.68	0.42	0.52
Lotte	0.82	**0.90**	**0.86**	0.70	**0.90**	0.78
Lotte-Base	0.96	0.29	0.45	0.84	0.26	0.40
+ OI	0.91	0.64	0.75	0.84	0.74	0.79
+ OI+otm	0.90	0.72	0.80	0.83	0.79	0.81
+ Fuzzy	0.88	0.83	0.85	0.79	0.84	0.81
+ Skip	0.85	0.84	0.84	0.75	0.85	0.79
+ Ellipsis	0.90	0.74	0.82	0.83	0.82	0.82

Table 9: Precision, recall, and F_1-score for *Die Judenbuche* and *Michael Kohlhaas*.

of Lotte, we analyze the different types of false positives as shown in Table 10. The second column shows the total number of false positives, followed by the counts for three relevant types of false positives. For example, out of the 279 false positive matches found by Lotte for *Die Judenbuche*, 64 are type *other*, that is, there is a match in our gold annotations which was not annotated as a quote from the primary literary work but some other text, for example, other literary works or scholarly works. For example, *Die Judenbuche* quotes the Bible and that same quote is quoted in a scholarly work and attributed to the Bible by our annotations but, of course, Lotte counts it as a match. 31 are of type *short*, that is, a match with five words or more was found but the corresponding gold item is only four words long. *O+S* is the combination of the two previous cases. The remaining false positives do not belong to any category.

Comparing the numbers for Copyfind, Lotte and SimT, we find that for both literary works, SimT and Copyfind have less false positives of the three types. Counting these as true positives Lotte's precision would improve relative to the other two approaches.

Another reason for the high number of false positives is that a large number of quotations are not annotated at all because they are not correctly highlighted (e.g., with quotation marks). This issue is worse for Lotte because of the improved handling of quotation-specific properties which leads to a higher number of false positives which are actually true positives but are missing in our data. The false positives which do not belong to any of the

Approach	Total		Other		Short		O+S	
	J	K	J	K	J	K	J	K
BLAST	227	646	30	37	33	45	0	1
Copyfind	174	232	46	22	24	29	0	0
Lotte	279	404	64	47	31	47	2	0
SimT	128	186	40	22	12	25	0	0
TextMatcher	130	112	14	1	4	13	0	0

Table 10: False positives counts for *Die Judenbuche* (J) and *Michael Kohlhaas* (K).

mentioned types (other, short and O+S) have an average length of 6.93 words (J) and 5.75 words (K). Of those matches, 45 (J) and 79 (K) are string equal when case is ignored. The average normalized Levenshtein distance of the source and target text string is 0.95 (J) and 0.92 (K). These results show that it is very likely that most of the false positives are not actually false positives.

5.3 Ablation Study

The presented approaches differ in the functionality they support as shown in Table 8. To evaluate the influence of the different functionalities, we compare the results of different versions of Lotte which emulate the absence of different functionality (cf. Table 9).

BLAST is optimized for fuzzy matching of OCRed text and allows for a high number of mismatched characters. This results in a high number of errors and makes it hard to link specific functionality to specific results. Therefore, BLAST will not

be considered in this comparison.

Lotte-(*base*) is Lotte with all five functionalities (cf. Table 8) disabled. This results in low recalls of 0.29 (J) and 0.26 (K). Lotte-(*OI*) is the base system with order independent matching enabled. This more than doubles the recalls to 0.64 (J) and 0.74 (K) and explains why TextMatcher has the lowest recall of all systems as it is the only system that does not support order-independent matching. SimT on the other hand only supports order-independent matching and achieves a rather high recall. Lotte-(*OI+otm*) is the OI-system with one-to-many matching added. This again improves recall significantly.

The last three systems Lotte-(*fuzzy*), Lotte-(*skip*), and Lotte-(*ellipsis*) are all based on Lotte-(OI+otm) with one functionality added. The improvement in recall for Lotte-(skip) explains the better performance of Copyfind over SimT.

Although around 16 % (J) and 19 % (K) (cf. Table 6) of quotations contain ellipses, the performance of Lotte-(ellipsis) is not a lot better. This might be because even without explicitly handling ellipses, a system will still find at least some part of the full match.

One more notable result is the high precision for all the different variants of Lotte. As discussed earlier, our data makes it hard to accurately evaluate the precision. We therefore decided to optimize for recall and assume a higher precision based on our findings in Section 5.2.

6 Visualizing and Exploring Key Passages

Here, we describe how the results of Lotte are integrated into an interactive website for visualizing and exploring key passages.[4]

6.1 Segmentation to Identify Key Passages

We process the output of Lotte to identify key passages by combining overlapping matches and generating minimal non-overlapping segments with frequency counts. Figure 2 sketches the segmentation process. The example contains the source text $w_1\ w_2\ \ldots\ w_9\ w_{10}$ and different sequences which quote the source to a varying extent. We segment the source text into non-overlapping segments and

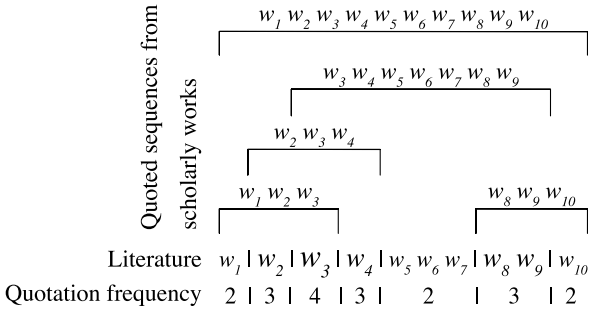

Figure 2: Visualization of the segmentation process.

count the frequency for each segment. For example, the sequence w_1 appears in two texts, sequence w_2 in three texts, sequence $w_5\ w_6\ w_7$ in two texts, and so on. This results in the quotation frequency shown at be bottom of Figure 2. The result of this segmentation process is used to visualize the literary text and the scholarly texts as described next.

6.2 Annette – A Visualization and Exploration Website

A screenshot of the website is shown in Figure 3. On the left, a heatmap of the complete literary text shows the distribution of quoted passages. The darker the text, the more often it has been quoted and thus the more important it is assumed to be. Next to the heatmap, the literary work is shown. The grayscale is determined by how many scholarly works quote some part of a key passage. That is, the color is always the same for the whole key passage. The font size is determined by how often a minimal segment is quoted. At the bottom, next to the literary text, a list of all scholarly works is shown. On the right, the top ten key passages are shown.

Starting from the initial screen, we can choose between different paths. The first option is to select a key passage by clicking on it. At the bottom, next to the literary text, a list of scholarly works which contribute to the selected key passage is then shown along with a preview of the quoted text. By clicking on one of the quoted texts, we can select a specific scholarly work. The text of that scholarly work is then shown at the top right. We can then go through that text and select other quoted passages. The bottom right shows how often the selected key passage was quoted and by how many scholarly works. Below, we can find the top ten most quoted segments of that passage. We can go back to the initial screen by clicking on the title at the top. From there, the other option is to select one of the

[4]The website is available at https://hu.berlin/annette-en. The source code of a white-label version is available at https://scm.cms.hu-berlin.de/schluesselstellen/lottevizex licensed under the Apache License 2.0.

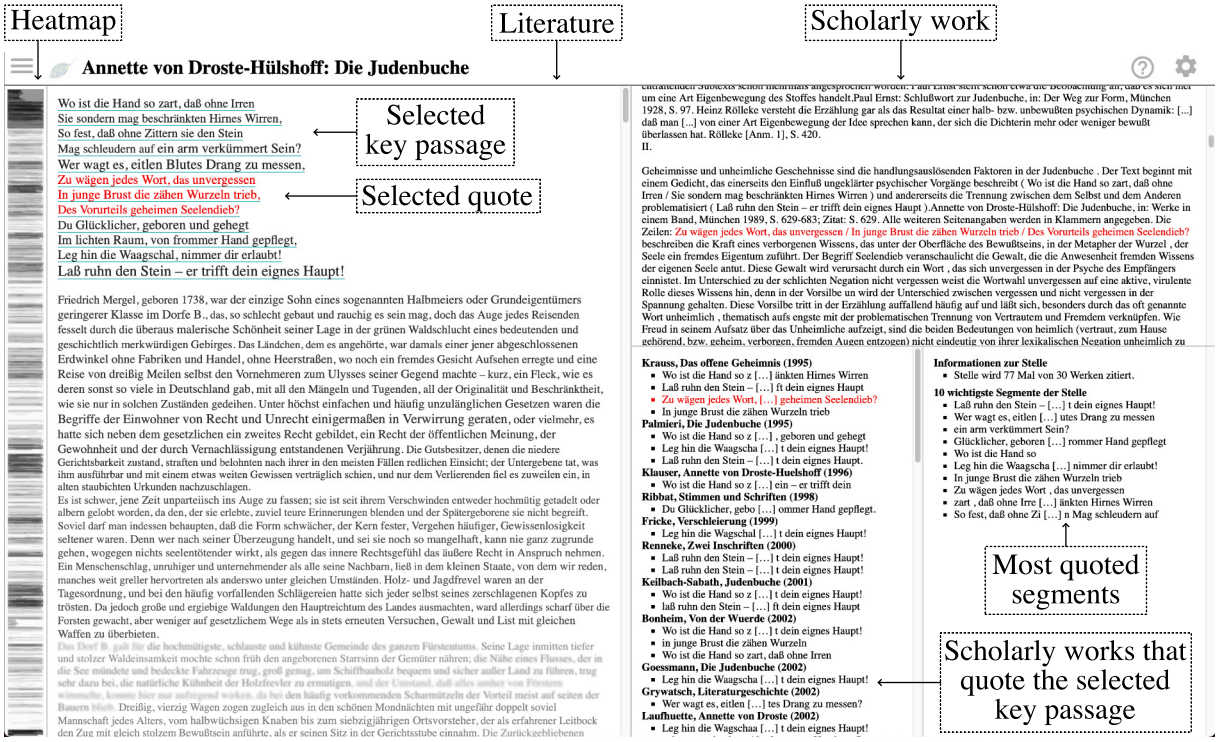

Figure 3: A screenshot of the website for visualizing and exploring key passages.

scholarly works from the list of all works. This will show the text of the selected work in the top right with all quotations highlighted.

7 Conclusion

We presented an approach for finding and visualizing key passages in literary works using scholarly works. For finding the quotations, we developed a system called Lotte by adapting Sim_text (Grune and Huntjens, 1989). Our approach outperforms prior approaches for text reuse detection. The matches are further processed to identify key passages by combining overlapping matches. We also presented Annette, a website that visualizes the literary work and scholarly articles together with the found quotations and thus allows us to explore the identified key passages and their origin. The current system only considers matches of length five and greater. In the future, we want to also identify shorter quotations and investigate how much information these add compared to longer ones. Another limitation of the current system is the missing suppport for handling ambiguous quotations. We have shown that this becomes more relevant the longer the source texts are. One solution to resolve such cases could be to utilize page references about the quoted passage, which are often included in the scholarly text. Furthermore, we aim to identify and analyze paraphrases and renarrations of literary works.

Acknowledgements

Parts of this research were funded by the German Research Foundation (DFG) priority programme (SPP) 2207 *Computational Literary Studies* project *What matters? Key passages in literary works* (grant no. 424207720).

References

Stephen F. Altschul, Warren Gish, Webb Miller, Eugene W. Myers, and David J. Lipman. 1990. Basic local alignment search tool. *Journal of Molecular Biology*, 215(3):403–410.

Lou Bloomfield. Copyfind [online].

Marco Büchler. TRACER: A text reuse detection machine [online]. 2016.

Annette von Droste-Hülshoff. 1979. *Die Judenbuche*. Insel Verlag, Frankfurt am Main.

Dick Grune and Matty Huntjens. Detecting copied submissions in computer science workshops [online]. 1989.

Timothy C. Hoad and Justin Zobel. 2003. Methods for identifying versioned and plagiarized documents. *J. Am. Soc. Inf. Sci. Technol.*, 54(3):203–215.

Sofia Kalaidopoulou. 2016. similarity texter: A text-comparison web tool based on the "simtext" algorithm. Bachelor's thesis, Hochschule für Technik und Wirtschaft, Berlin. Source code available at https://people.f4.htw-berlin.de/~weberwu/simtexter/app.html.

Heinrich von Kleist. 1978. Michael Kohlhaas. In Michael Holzinger, editor, *Werke und Briefe in vier Bänden*, pages 7–113. CreateSpace Independent Publishing Platform.

Vladimir I. Levenshtein. 1966. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, 10(8):707–710.

Derek Miller. To Quote or Not to Quote [online].

C. D. Paice. 1980. The automatic generation of literature abstracts: An approach based on the identification of self-indicating phrases. In *Proceedings of the 3rd Annual ACM Conference on Research and Development in Information Retrieval*, SIGIR '80, page 172–191, GBR. Butterworth & Co.

Jonathan Reeve. Jonathanreeve/text-matcher: First zenodo release [online]. 2020. version 0.1.6.

M. Scott and C. Tribble. 2006. *Textual Patterns: Key Words and Corpus Analysis in Language Education*. Studies in corpus linguistics. J. Benjamins.

Malcolm Slaney and Michael Casey. 2008. Locality-sensitive hashing for finding nearest neighbors. *IEEE Signal Processing Magazine*, 25(2):128–131.

David A. Smith, Ryan Cordell, Elizabeth Maddock Dillon, Nick Stramp, and John Wilkerson. 2014. Detecting and modeling local text reuse. In *Proceedings of the 14th ACM/IEEE-CS Joint Conference on Digital Libraries*, JCDL '14, page 183–192. IEEE Press.

TEI Consortium, eds. TEI P5: Guidelines for electronic text encoding and interchange [online].

Aleksi Vesanto, Filip Ginter, Hannu Salmi, Asko Nivala, and Tapio Salakoski. 2017a. A system for identifying and exploring text repetition in large historical document corpora. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 330–333, Gothenburg, Sweden. Association for Computational Linguistics.

Aleksi Vesanto, Asko Nivala, Heli Rantala, Tapio Salakoski, Hannu Salmi, and Filip Ginter. 2017b. Applying BLAST to text reuse detection in Finnish newspapers and journals, 1771-1910. In *Proceedings of the NoDaLiDa 2017 Workshop on Processing Historical Language*, pages 54–58, Gothenburg. Linköping University Electronic Press.

Simone Winko. The making of plausibility in interpretive texts. Analyses of argumentative practices in literary studies [online]. 2017–2020. DFG-funded research project (grant no. 372804438).

Jin-ge Yao, Xiaojun Wan, and Jianguo Xiao. 2017. Recent advances in document summarization. *Knowledge and Information Systems*, 53(2):297–336.

Using Referring Expression Generation to Model Literary Style

Nick Montfort
Massachusetts Institute of Technology
Cambridge, MA, USA
nickm@nickm.com

Ardalan SadeghiKivi
Massachusetts Institute of Technology
Cambridge, MA, USA
ardalan@mit.edu

Joanne Yuan
Massachusetts Institute of Technology
Cambridge, MA, USA
joanneyu@mit.edu

Alan Y. Zhu
Massachusetts Institute of Technology
Cambridge, MA, USA
alanyzhu@mit.edu

Abstract

Novels and short stories are not just remarkable because of what events they represent. The narrative style they employ is significant. To understand the specific contributions of different aspects of this style, it is possible to create limited symbolic models of narrating that hold almost all of the narrative discourse constant while varying a single aspect. In this paper we use a new implementation of a system for narrative discourse generation, Curveship, to change how existents at the story level are named. This by itself allows for the telling of the same underlying story in ways that evoke, for instance, a fabular or parable-like mode, the style of narrator Patrick Bateman in Brett Easton Ellis's *American Psycho*, and the unusual dialect of Anthony Burgess's *A Clockwork Orange*.

1 Introduction

It is well-known in narrative theory (narratology) that narratives achieve their power not only because of *what* they represent but also because of *how* they represent it. Narrative theorists generally agree that there is utility in at least conceptualizing a story level (the "content") separate from the discourse level (the "expression").

With awareness of this separate narrative discourse, we can ask how the expression particularly contributes to the effect of a novel or short story. For instance, Herman Melville's *Moby-Dick* is a narrative of personal experience told by Ishmael. Imagine that all the events of the novel remained the same, it still focalized Ishmael, but the narration was like that of Hemingway's *The Old Man and the Sea*. Ishmael would no longer be the "I" of the story and the book would no longer be told in his characteristic voice or enunciation. The narrator would be rather covert and unmarked. How would the effect be different? Or, what if Ishmael remained the "I" and spoke as he does in the novel, but was not on the Pequod — another character who had all the same experiences took his place on the whaling voyage — and so the narrator had his distinctive voice, but without being a character in the diegesis?

Researchers have investigated individual novels using statistical methods that have more typically been applied to large corpora of literature (Clement, 2008; Kelleher and Keane, 2017; Wang and Iyyer, 2019). Hypothetical and speculative questions of the sort we pose above, about literary style and the style of individual novels, can be addressed in a different way, by modeling narrative style computationally. Generating variations of a large classic novel is not a reasonable goal. By creating small-scale narratives, however, more akin to folktales or conversational stories but in different literary styles and in fine-grained genres, we can try out different ways of telling the same underlying story to see what literary effects can be produced using a compact model.

Building on our earlier work generating different narrative styles, in this study we focus on one type of variation, corresponding to a classic area of natural language generation research, referring expression generation (REG). That is, we look at how the narrator names existents (those entities that exist) in the story, corresponding to characters, places, and other things or objects. Our question is, when we focus on the naming of existents, allowing few other variations, how much influence can we exert over literary style? Which literary styles can we computationally model in this way?

Unlike almost all work on literary study done with computational linguistics approaches, ours does not involve automatic feature extraction, stylistics, or the annotation of a corpus. We devise our symbolic models of narrative style "manually" by specifying the possible vocabulary that can be used and high-level narrative parameters.

Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH), pages 64–74
December 19, 2021.

2 A Style Generator, Not a Story Generator

A story or plot generator is a computational procedure, a set of instructions that, when applied to given inputs, produces an output considered a story. These generators work at the level of content (the story level). Curveship is focused on the narrative discourse and is *not* a story or story-level generator.

An important early story generation project is TALE-SPIN (Meehan, 1997), operating on story data similar to that in (Klein et al., 1973) to produce animal stories. It simulates reasoning and behavior in a virtual world, using planning to find how the characters can accomplish (or fail at) their goals. Along these lines, TV melodrama system UNIVERSE (Lebowitz, 1985) has a character creation cycle and simulates each personal life. MINSTREL (Turner, 1994) tells stories about King Arthur's court, operating at two different levels of author and actors goals, using a planning stage and a problem-solving stage. Many earlier projects along with FABULIST (Riedl and Young, 2010) refer to an explicit model of narrative or, in the case of MEXICA (Pérez y Pérez, 1999), one of the creative writing process.

These related projects do not focus on the generation of the *narrative discourse*, as Curveship does. Because both levels are important to narrative, two collaborations have already been done to integrate the system with MEXICA (Montfort, 2009; Montfort et al., 2013).

A system operating at the expression level is PAULINE (Hovy, 1988), generating appropriate texts from a single representation and accounting for the hearer's sensitivity and contexts of communication. Along similar lines, but working in creative and narrative contexts, are systems for dialog generation, including (Hämäläinen and Alnajjar, 2019). Curveship differs from both in that it is specialized to deal with narration, not pragmatics or dialog. Of recent work, that on style transfer, e.g. (Reif et al., 2021), is related, although in this case we see a diametrically opposite approach of using large, opaque language models and not even having a "source text" in a particular style. While a provocative direction and one we hope could inform narrative studies, our approach is to use a highly parsimonious model in which all narrative parameters are explicit. Our goal is the study and human understanding of style, rather than the efficient and general automated transfer of it.

3 Stylistic Variation in Literature and Narratology

Narrative and other stylistic variation has been explored for centuries without computing, long before narratology. Erasmus's 1512 *Copia*, in Latin, systematically shows how to embellish, expand, and vary speech and writing, explaining that expression as well as subject matter should be "abundant" (Erasmus, 2007). His variations include syntactical ones along with the use of synonyms, heterosis, enallage, and other figures of speech.

A creative work more specific to narrative is *Exercises in Style,* first published in French in 1947 (Queneau, 1981). It narrates the same uninteresting events 99 times. Only the style engages the reader, who finds a quotidian story told as a book blurb, an entire three-act play, a haiku, etc. Each section has a practical and frank title, e.g., "Notation," "Official Letter," "Comedy." Some variations are specific to narrative, as with "Retrograde," in which events are represented in reverse chronological order. There are others that are not, including lexical variations in which the story is narrated in anagrams and permutations along with variations in which the text is dismantled and listed by the different parts of speech.

Queneau's book inspired many others, including the graphic novel *99 Ways to Tell a Story* (Madden, 2006). Although the particular banal sequence of events is different, the framework is the same, exploring the possibilities of the comic medium rather than text. The visual narratives include a map, a how-to-draw process, a public service announcement, an advertisement, and an image in the style of the Bayeux tapestry. Some are in the style of famous comic artists.

More theoretical attempts to systematically understand how the narrative discourse can vary, independent of the story or content, have been expressed in the *narratologies* or specific narrative theories of different scholars since the 1960s, e.g., (Bal, 1985; Chatman, 1980; Rimmon-Kenan, 1983) and new editions thereof. The system we have developed and use in this study is based mainly on (Genette, 1979) as revised in (Genette, 1988), with awareness of other aspects of narrative pointed out in (Prince, 1982) and other sources.

Narratology began with the the novel and drew from linguistic ideas but is now clearly transmedial (Ryan et al., 2004) and, in dealing with all sorts of representations of events, is not limited to literary

or linguistic sorts of study. Nevertheless, if it is considered in a linguistic framework it must be acknowledged to operate at the discourse level, where for instance intersentential reference is a concern.

4 Relevant Work on Referring Expression Generation

In a narrative framework, and focusing on text, we consider a referring expression to be any noun phrase or substitute for an NP that designates an existent. Our definition, limited to narrative, is consistent with standard ones, presented when the task is to refer to objects in a blocks world (Winograd, 1972) or in photographs of the environment (Zarrieß and Schlangen, 2016). Our research does not involve such representations, but our system nevertheless needs to convert non-linguistic information (the story-level "content") to natural language, and in this regard (Krahmer and van Deemter, 2012) it undertakes this REG task.

In the context of our work so far on very short stories, we are not mainly concerned with REG that selects from available attributes to produce NPs with appropriate modifiers, a common goal of REG research. Instead, we are concerned with adherence to systematic naming conventions of different sorts.

We use REG in a contained and tractable framework in which existents and events are symbolically modeled and the narrative style is, similarly, completely and formally defined. Our approach is not corpus-based and our system, rather than being trained on any data, uses human-authored expressions, selected and combined computationally. Our project makes use of very established REG techniques to investigate literary style.

5 Our Platform for Narrative Modeling

We use a system developed originally in Python to implement parser-based interactive fiction (Montfort, 2007, 2009), now in a new JavaScript formulation. The current Curveship-js allows for teaching and creative work as well as research. Students have used it to learn about narrative theory concepts. Although finished creative works have not yet been published using the system, it has seen some use by writers and artists.

Curveship-js allows for a wide variety of changes to be automatically made at the level of narrative discourse, including changes to the order of events in the telling, ellipsis, focalization, time of narrating, and the "I" and "you" of the narrative (the clearest signs of narrator and narratee). Using a simple underlying representation, Curveship-js generates text and determines grammatical specifics according to high-level narrative parameters.

To pursue this current project, we implemented a clean split between story (the "content") and narrator, with each level modeled in a different files and with the ability to associate arbitrary numbers of narrators with the same story. As much linguistic information as possible has been eradicated from the story-level representation, so that now narrators not only have different global or general parameters but also each have their own names for existents and their own verb phrases associated with events.

6 Referring Expressions and Literary Style

Of many compelling literary styles and methods of naming existents, we have identified a few for initial modeling. We test our idea that these styles are distinctive because of how referring expressions are used. We have selected literary styles which present different sorts of challenges. Some have only subtle differences. Others use very marked and elaborate styles of naming.

References to characters using an NP are in bold. Similar references to other existents are in Italic. All pronominal references remain in Roman.

6.1 Parable or fable

Uses very generic names, identifying individuals by at most profession or with a single descriptive adjective, sometimes even less.

> Again, the *kingdom of heaven* is like unto *treasure* hid in a *field*; the which when **a man** hath found, he hideth, and for joy thereof goeth and selleth all that he hath, and buyeth *that field.* (Matthew 13:44, KJV)

> **The fox** who longed for *grapes*, beholds with pain / The tempting *clusters* were too high to gain; (Dudley, 1970)

> Go and visit **grandmother,** who has been sick. Take her *the oatcakes* I've baked for her on *the hearthstone* and a little *pot of butter*. **The good child** does as **her mother** bids ... (Carter, 1979)

Note that no characters at all are named in Carter's "fable." It is based on "Little Red Rid-

ing Hood," which fits, in terms of genre, in 6.3, but is in the *style* described here in 6.1.

6.2 Literary simplicity

Evoking the parable or fable, some literary writing has extremely straightforward syntax and uses simple, generic expressions for characters, mentioning proper names only rarely.

> He was **an old man** who fished alone in *a skiff* in *the Gulf Stream* and he had gone eighty-four days now without taking a fish. In the first forty days **a boy** had been with him. (Hemingway, 2003)

A syntactically simple run-on sentence here is characteristic of Hemingway's style. While the old man's name, Santiago, is mentioned in this novel, almost all references to the character are "the old man" or use a pronoun.

> **The boy** leaned on *the cart* and adjusted *the wheel*. What do you see? **the man** said. Nothing. He lowered *the glasses*. It's raining. Yes, **the man** said. I know. *McCarthy:2006*

Unmarked direct discourse (speech that is not surrounded by quotation marks) is seen in this extract, and is characteristic of McCarthy's style. Again, of the very syntactically simple sentences, one is a run-on with two main clauses joined by "and." The duo encounter a named character, Ely, but neither of the main characters' names are mentioned. The differences in style from 6.1 Parable or fable here are subtle. That style actually often includes more complex syntax, however, and completely eschews proper names.

6.3 Folktale or fairy tale

Oral versions of these were significantly different, but we are considering literary style. The main character is often identified by a short sort of proper name or nickname, not including a family name: Cinderella, Snow White, Rapunzel, Little Red Riding Hood. This main character's name is of course also the name by which the story is known. Subsidiary characters are often named more generically according to their roles: The prince, the Big Bad Wolf. Literary works that draw on folklore sometimes use similar naming conventions alongside metafictional (self-conscious and self-referential) writing techniques.

> All on *one summer's day,* **the King of Hearts** calls for *the tarts* baked for him by **his Queen,** only to find they have been stolen. All fingers point to his former page, the **Knave of Hearts** ... (Coover, 2005)

6.4 Initials only for some existents

At various points in literary history authors have chosen to indicate existents (characters, villages, etc.) using only their first initials, sometimes with a long dash, period, or other special typographical intervention after them. It is possible this allows a narrative to be read in a way that is more general and universal; it may also suggest that the narrative is about real-world existents the narrator is choosing to anonymize.

> In my return from Italy I brought him with me to the country in whose language he had learn'd his notes—and telling the story of him to **Lord A—Lord A** begg'd *the bird* of me—in a week **Lord A** gave him to **Lord B—Lord B** made a present of him to **Lord C**—and **Lord C**'s gentleman sold him to **Lord D** for a shilling—**Lord D** gave him to **Lord E**—and so on—half round the alphabet... (Sterne, 1986)

> In this, I am bound to say, **Mr. A.** was but sustaining the tradition conceived originally by his predecessor, **Mr. P.**, a Harvard man, who until his departure from *Vingt-et-Un* succeeded in making life absolutely miserable for **B.** and myself. (Cummings, 1949)

> I am a well-known folklorist, an authority on the *A———s*, a tribe I have no intention of disgracing by my interest. (Cohen, 1993)

6.5 The novel of manners

While literary writing is often concerned with characters' place in society, this is magnified in the novel of manners. Proper names are used with courtesy titles, which are repeated, and referring expressions otherwise relate to social status.

> Not all that **Mrs. Bennet**, however, with the assistance of her **five daughters,** could ask on the subject, was sufficient to draw from **her husband** any

> satisfactory description of **Mr. Bingley**. They attacked him in various ways; with barefaced questions, ingenious suppositions, and distant surmises; but he eluded the skill of them all, and they were at last obliged to accept the second-hand intelligence of **their neighbour Lady Lucas**. Her report was highly favourable. **Sir William** had been delighted with him. (Austen, 2008)

6.6 Brand names for objects

An obsessive use of brand names and mention of the places where items were purchased can indicate a consumer fixation or even psychopathology.

> He continues talking as he opens *his new Tumi calfskin attaché case he bought at D. F. Sanders.* He places *the Walkman* in the case alongside *a Panasonic wallet-size cordless portable folding Easa-phone* (he used to own *the NEC 9000 Porta portable*) and pulls out *today's newspaper*. (Ellis, 1991)

6.7 Dialect and idiolect

Narration is sometimes done using idiosyncratic lexical or vocabulary choices. These can include misspellings and generally infelicitous word usage. An invented dialect can also be employed, perhaps to project science-fictional worlds with an uncanny relationship to our own.

> I took *the large moloko plus* to *one of the little cubies* that were all around *this mesto,* there being like *curtains* to shut them off from *the main mesto* ... and then there were colours like no one had ever viddied before, and then I could viddy like *a group of statues* a long long way off ... (Burgess, 1986)

7 The Example Story and Corresponding Narrative Specifications

A very short story, "First Class," was modeled at the content level in Curveship-js. While "First Class" is an original story, it was inspired by part of the 15th track of the album *Black on Both Sides* by Yasiin Bay (previously known as Mos Def), which describes a racial microagression not present in our story. We encourage those consulting this research to listen to this song.

The underlying story model has no linguistic information attached to it that is used in realization. The existents and events do have tags, for internal use: A Curveship-js author or researcher uses these to connect the story-level existents and events to discourse-level names and verb phrases. In "First Class," the existents are two Places (gate and firstClass), four Actors (the class for modeling characters; these are celebrity, gateOfficial, flightAttendant, and passenger), and eleven Things including boardingPass, scanner, seat1A, seat1B, notepad, pen, and some items of clothing.

There are also sixteen (16) events. The story file that includes these existent and event representations, excerpted in figure 1, is concise (less than 3KB).

To produce different specific narratives, a narrator file is used for each. Although Curveship-js has many capabilities for varying the narrative, we have limited our use of the system so that we strongly emphasize only one: the facility for different naming or generation of referring expressions. To generate a few of the final results, we also vary the verb phrases used to produce representations of events, and we also make other minor changes in time of telling, ellipsis, focalization, and the like.

One does not need to define all names for existents in a particular narrator file. If an entity is a character (Actor) who is female and a child, for instance, Curveship-js will automatically generate "a girl" as a name for the character on first reference, and "the girl" thereafter, by instantiating the GenericName class. Or, using the Names class, one can define a simple, short name with an initial NP to be used and (optionally) an NP to be used on subsequent reference. There are extensions of this to allow for Actors with first and last names and courtesy titles (ProperNames) and for Things that are indicated by trade name and their place of purchase (BrandNames).

The system is a free/libre/open-source software system, so others may use it to reproduce our results or as the basis for their own research or even creative projects. (Montfort and contributors, 2021)

8 Generated Results

The following are all outputs from the system, each a different version of "First Class" that attempts to imitate a narrative style in section 6.

```
actor.celebrity = new Actor(place.gate, ″male″);
actor.gateOfficial = new Actor(place.gate, ″female″); ...
thing.boardingPass = new Thing(actor.celebrity);
thing.boardingPass.owner = actor.celebrity;
thing.scanner = new Thing(place.gate);
thing.jacket = new Thing(actor.celebrity); ...
// EVENTS
ev.scan = new Event(actor.celebrity, thing.boardingPass, temporal.against, thing.scanner);
ev.beep = new Event(thing.scanner);
ev.gasp = new Event(actor.gateOfficial);
ev.pocket = new Event(actor.celebrity, thing.boardingPass, temporal.in, thing.jacket);
ev.pocket.alters(thing.boardingPass, ″location″, actor.celebrity, thing.jacket);
```

Figure 1: An excerpt from the story file for "First Class."

8.1 Parable or fable

A man walked from the gate. He went to the first class cabin. He sat in a wide seat. A woman was surprised. A flight attendant noticed the man's sneakers. He said "sir, I believe you're in the wrong cabin" to the man. The man took his boarding pass from his jacket. He showed his boarding pass to the flight attendant. The flight attendant apologized. He said "I shouldn't ask, but ... my daughter would really love to have your autograph" to the man.

8.2 Literary simplicity

The scanner beeped. The airline employee gasped. Harry had held his boarding pass against the scanner. He departed the gate. He walked to the first class seating area. He sat in a seat in the first row. A woman expressed a bit of surprise. A flight attendant noticed the celebirty's sneakers. He said "sir, I believe you're in the wrong cabin." The celebrity took his boarding pass from his jacket. He showed his boarding pass to the flight attendant. The flight attendant muttered "oh, I'm sorry." He procured a pen and a notepad. He said "I shouldn't ask, but ... my daughter would really love to have your autograph."

8.3 Folktale or fairy tale

One day... The Frog Prince held his boarding pass against a panel on the airport turnstile. It beeped. A woman gasped. The prince placed his boarding pass in his jacket. He walked from the gate. He entered to the first class cabin. He sat in a comfortable seat. A woman puckered up involuntarily. A flight attendant noticed the prince's sneakers. He said "sir, I believe you're in the wrong cabin." The prince took his boarding pass from his jacket. He showed his boarding pass to the flight attendant. The flight attendant apologized by saying "oh, I'm sorry." He obtained a pen and a notepad. He said "I shouldn't ask, but ... my daughter would really love to have your autograph." The end.

8.4 Initials only for some existents

Mr. S——— holds his boarding pass against a scanner. The scanner beeps. A lady working for the airline gasps. Mr. S——— places his boarding pass in his sportscoat. He departs the gate. He walks to first class. He sits in seat 1B. A young female executive reacts. A flight attendant notices Mr. S———'s kicks. He sneers "sir, I believe you're in the wrong cabin" to Mr. S———. Mr. S——— gets his boarding pass from his sportscoat. He shows his boarding pass to the flight attendant. The flight attendant mutters "oh, I'm sorry" to him. He grabs a pen and a notepad. He says "I shouldn't ask, but ... my daughter would really love to have your autograph" to Mr. S———.

8.5 The novel of manners

An airline employee gasped. Sir Harry Styles had held his boarding pass against a scanner. He placed his boarding pass in his sportscoat. He departed the turnstile. He walked to the first class cabin. He sat in seat 1B. Ms. Carly Fiorina reacted. A flight attendant noticed Sir Styles's casual shoes. He sneered "sir, I believe you're in the wrong cabin" to Sir Styles. Sir Styles got his boarding pass from his sportscoat. He showed his boarding pass to the attendant. The attendant muttered "oh, I'm sorry" to him. He grabbed a pen and a notepad. He said "I shouldn't ask, but ... my daughter would really love to have your autograph" to Sir Styles.

```
names.gate = new Names(″the gate″);
names.firstClass = new Names(″the first class cabin″); ...
names.seat1A = new Names(″a wide seat″); ...
vp.depart = new VerbPh(″walk from″);
vp.board = new VerbPh(″go″);
vp.sit = new VerbPh(″sit″);
vp.beSurprised = new VerbPh(″is surprised″);
```

Figure 2: An excerpt from the "parable teller" narrator file for "First Class."

8.6 Brand names for objects

The type of guy who can get a reservation at Le Bernardin walks to first class. He sits in seat 1B. I notice coolly. A male flight attendant glances at the famous guy's Air Jordan 4 Retro Kaws purchased from Flight Club. He sneers "sir, I believe you're in the wrong cabin" to the famous guy. The famous guy gets his boarding pass from his bespoke Michael Andrews sportscoat. He shows his boarding pass to the male flight attendant. The male flight attendant mumbles "oh, I'm sorry" to him. He pulls out a BIC pen from K-Mart on Astor Place and a Mead memo pad bought at Key Foods. He says "I shouldn't ask, but ... my daughter would really love to have your autograph."

8.7 Dialect and idiolect

This uniformed devotchka gasped. Sir Harry Styles had held his boarding pass against a scanner. He placed his boarding pass in his carman. He walked to the first class cabin. He sat in seat 1B. This forella reacted. A veck viddied Sir Styles's sabogs. He sneered "sir, I believe you're in the wrong cabin" to Sir Styles. Sir Styles got his boarding pass from his carman. He showed his boarding pass to the veck. The veck muttered "oh, I'm sorry" to him. He grabbed a pen and a notepad. He skazated "I shouldn't ask, but ... my daughter would really love to have your autograph" to Sir Styles.

9 Discussion

We refer to styles presented in section 6 and modeled in section 8 as **1–7**. While results varied, we attempted to model these seven types of narrative style seriously and the process has provided us with some insights into what aspects of linguistic and narrative representation are most important to each sort of style. In other words, the system has helped us think about style in ways that complement other inquiries.

This method has allowed us to see how some styles are more easily imitated, or at least signaled, by varying the way referring expressions are generated, while others (if they are in fact best considered distinct styles) will require different sorts of intervention. Perhaps we have been able to discern that some styles are simply more straightforward while others — forgive us — are more hairy.

Rather than discuss these in seven sections, we offer a synthetic and comparative discussion of the ways narrative style has been modeled and the text that resulted.

There are obviously many aspects of style we have not yet computationally modeled. We tried to avoid translated texts, but many canonical examples of **1** Parable or fable are in translation. The first quotation is of very archaic English (translated from Koine Greek) and the second in verse — this time in translation from Ancient Greek. Carter's fable, on the other hand, is in prose and in contemporary English. To present work in the style of parables from the King James Bible or from 17th Century translations of Aesop's Fables, finer distinctions must be made.

Although our focus has been on referring expressions, we found it necessary to use ellipsis, change the order of events in the telling, change the time of speaking, and in one case (8.6) focalize a particular character in order to do a reasonable job of modeling style. We also changed the way that events were represented by having our narrators use different verb phrases, not just their own names for Existents. This was particularly important in 8.7, where the dialect includes unusual verbs.

We found that we were able to do reasonable work matching only three styles when strictly chronological narration was used, even in the limited context of this simple underlying story. The ones that seemed apt when a chronological ordering was used were **1** Parable or fable, **3** Folktale or fairy tale, and **6** Brand names for objects. The first two styles are associated with orature and with

```
names.firstClass = new Names(″the first class cabin″); ...
names.gateOfficial = new Names(″this uniformed devotchka″);
names.celebrity = new ProperNames(″Harry″, ″Styles″, pronoun.masculine, ″a celebrity″, ″Sir″); ...
names.boardingPass = new Names(″a boarding pass″);
names.scanner = new Names(″a scanner″); ...
vp.scan = new VerbPh(″hold″);
vp.gasp = new VerbPh(″gasp″);
vp.pocket = new VerbPh(″place″);
vp.board = new VerbPh(″walk″);
```

Figure 3: An excerpt from the "Alex Delarge" (narrator of *A Clockwork Orange*) narrator file for "First Class."

fairly simple and short stories, so it is not surprising that they can be told in chronologically straightforward ways and have their style remain identifiable. The last is perhaps most interesting, as it involves complex naming and concern with not only branding but also purchase history. That the narration is flat and direct in other aspects may help to highlight the narrator's fascination, or indeed obsession, with consumerism.

6 Brand names for objects was also the only style in which it seemed important to strictly focalize a single character, the woman passenger on the plane whose seat is next to the celebrity's. Although **7** Dialect and idiolect is also associated with narratives of personal experience, the unusual enunciation of the diegetic narrator is so distinctive that it seems admissible to have the narration include events that this narrator may not have seen.

While **2** presents a style similar to that of Hemingway in *The Old Man and the Sea* or McCarthy in *The Road*, there are some noticeable differences. Simply in terms of referring expressions, it would be an improvement for the main character's name to be mentioned later in the discourse. Typographically, McCarthy employs unmarked direct discourse, foregoing any quotation marks. Hemingway uses quotation marks to indicate speech, but presents the main character's thoughts in free indirect discourse. While this does not end up confusing readers, it can mean that when beginning to read a sentence, it is not immediately clear whether the words are being narrated or are spoken by a character, adding some interest and complexity to the reading process. Adding this capability to our system would of course be useful.

In both of those "simple" books, even restricting our examination to representations of action rather than description or exposition, we find long sentences that are beyond what our system can currently generate, for instance: "He settled comfortably against the wood and took his suffering as it came and the fish swam steadily and the boat moved slowly through the dark water." (Hemingway, 2003) This is a run-on sentence with four parts, three of which would be straightforward to model using Curveship-js. It would be an improvement to be able to produce run-on sentences of this sort to create a flow or press of several actions that are not punctuated.

In **3** Folktale or fairy tale, a story "preface" and "postface" of one fixed sentence each was significant to signaling the style or genre. The tone that was generated is even more outrageous than in many of Robert Coover's stories in this style, because the Frog Prince immediately is involved in the quotidian, contemporary bureaucracy of having his boarding pass scanned. The moral of this seems to be that this "style" cannot always be generated at the level of narrative discourse, as it sometimes relates to the underlying content.

In **4** Initials only for some existents, the effect of this one stylistic change seemed most obscure or oblique. This may be because of the very wide range of contexts in which this sort of naming has been used. Our generated text reads like the production of a contemporary writer who might be trying to imitate a style from centuries ago, but without really understanding the relevant aspects of any particular style. Perhaps this is no surprise, as our "initials only" examples in 6.4 were from the 18th Century, the early 20th Century, and the late 20th Century. It seems that it is simply not enough, or may end up appearing to be an affectation, to alter only this way of making reference to existents.

There are certainly some limitations to the way the style of the novel of manners **5** is modeled by the system. The use of courtesy titles and proper names for supposedly important characters, while

others of lower class are referred to by role, does make this text consonant with *Pride and Prejudice* and similar works of fiction. We miss, however, certain aspects of framing and the sort of abstract declarations that Austen makes to inform the reader about the social world. These elements are not as overt in **1**, **2**, and **3**, for instance.

In **7** Dialect and idiolect, the incorporation of distinctive nouns and verbs certainly signals the style we are hoping to imitate — anyone familiar with *A Clockwork Orange* will be unable to avoid noticing that this narrative is in the style of the novel. While it uses these distinctive words to make a connection, there are nevertheless noticeable failings in the way text is generated. Alex interjects "like" as well as phrases such as "O my brothers and only friends" throughout the novel, interjections which seem to us important to the style and conspicuously missing from the generated text. He narrates what he is feeling and thinking; It is important to the style that the narrative is internally focalized. He negatively evaluates certain events and positively appraises others, for instance as "real horrorshow."

Our system does not currently have the ability to paraphrase utterances in direct discourse. Although parables and fables sometimes include direct discourse, it may be better paraphrased in 8.1 and probably in 8.3. Paraphrase would give the opportunity for additional dialectical narration in 8.8. Generally, the extent to which speech is directly quoted or is paraphrased seems important to literary styles such as these.

We believe 8.6 and 8.7 are most remarkable and easily identifiable as examples of particular literary styles. Rather than imagining that we were particularly good at modeling styles **6** and **7**, we take this as evidence that the original literary styles are so distinctive that fairly simple gestures toward them can indicate them.

We have omitted some uses of referring expressions that we have observed in literary work. For instance, we know of one case in a very short literary story in which only pronouns are used to refer to characters (Eason, 1992). Generally, better control over pronominalization (improving our current, primitive algorithm) would improve our ability to model styles, because some styles, as seen in **2**, are extremely spare, even to the extent of leaving some initial ambiguity in reference. For instance, to aid in modeling styles similar to **2** we would like to be able to introduce a character using a generic name or even a pronoun and, via cataphora, give the character's proper name later.

10 Future Work

Rather than try to draw more specific conclusions, we have chosen to identify next steps that would be productive for us and others seeking to computationally model narrative style.

Part of our project of modeling narrative style involves testing to what extent it is sensible, following narrative theory, to consider the underlying content or storyworld separately from the expression or narrative discourse. Future work should involve models of each level being further ramified. If a unified model could be developed by others that is simpler and more powerful than our two-level model, this would argue against the fundamental model narrative theory posits.

Some questions to answer, then, are whether a very wide variety of styles can be parsimoniously modeled using a single underlying story representation. Of course, expanding the number of styles modeled will be one direction for future work. A very broad investigation could involve taking on the "Queneau challenge" and producing 99 narratives that parallel Queneau's in *Exercises in Style*. As with our tentative research here, we would learn from this study which of the styles can be produced distinctively even with a limited model and which require more elaborate modeling.

An in-depth comparative analysis of a few important literary styles would also be an important direction for further research. Just as an attempt to address the "Queneau challenge" would encourage work on broad coverage, this effort would compel detailed study as the research digs deeply into a few important styles. We would need to generate lengthier stories in which reference is made to existents in different contexts.

Finally, other good evidence for the ability to usefully separate the content and expression levels would come from multi-lingual generation. We aim to generate narratives in different natural languages, initially, working with collaborators with appropriate expertise and adding the ability to generate in one other language. This will help us further develop a story/content representation that is not dependent on the particularities of expression. If we accomplish this, we should also be able to generate styles characteristic of different world literatures and gain insight into style across languages.

References

Jane Austen. 2008. *Pride and prejudice*. Oxford University Press, Oxford, England.

Mieke Bal. 1985. *Narratology: Introduction to the theory of narrative*. University of Toronto Press, Toronto.

Anthony Burgess. 1986. *A clockwork orange*. Norton, New York.

Angela Carter. 1979. The werewolf. In *The bloody chamber*. Harper —& Row, New York.

Seymour Chatman. 1980. *Story and discourse: Narrative structure in fiction and film*. Cornell University Press, Ithaca, New York.

Tanya E Clement. 2008. 'a thing not beginning and not ending': using digital tools to distant-read gertrude stein's the making of americans. *Literary and Linguistic Computing*, 23(3):361–381.

Leonard Cohen. 1993. *Beautiful losers*. Vintage Books, New York.

Robert Coover. 2005. Heart suit. In *A child again*. McSweeney's, San Francisco, California.

E. E. Cummings. 1949. *The enormous room*. The Modern Library, New York.

Aesop; Aphra Behn; Francis Barlow; Thomas Philipot; Robert Codrington; Thomas Dudley. 1970. *Æ*. H. Hills for Francis Barlow, London.

Bruce Eason. 1992. The appalachian trail. In *Flash fiction: 72 very short stories*. Turnstone Press, Winnipeg, Canada.

Bret Easton Ellis. 1991. *American psycho*. Vintage Books, New York.

Desiderius Erasmus. 2007. *On copia of words and ideas*. Marquette University Press, Milwaukee, Wisconsin.

Geérard Genette. 1979. *Narrative discourse*. Blackwell, Oxford.

Geérard Genette. 1988. *Narrative discourse revisited*. Cornell University Press, Ithaca, New York.

Mika Hämäläinen and Khalid Alnajjar. 2019. Creative contextual dialog adaptation in an open world rpg. In *Proceedings of the 14th International Conference on the Foundations of Digital Games*, pages 1–7.

Ernest Hemingway. 2003. *The old man and the sea*. Scribner, New York.

Eduard H. Hovy. 1988. *Generating Natural Language under Pragmatic Constraints*. Lawrence Erlbaum Associates, Hillsdale, New Jersey.

Conor Kelleher and Mark Keane. 2017. Plotting Markson's "mistress". In *Proceedings of the Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 33–39, Vancouver, Canada. Association for Computational Linguistics.

Sheldon Klein, J. F. Aeschlimann, D. Balsiger, Steven L. Converse, Claudine Court, Mark Foster, Robin Lao, J. Oakley, and Joel Smith. 1973. *Automatic Novel Writing: A status report*. Computer Science Department, The University of Wisconsin, Madison, Wisconsin.

Emiel Krahmer and Kees van Deemter. 2012. Computational generation of referring expressions: A survey. *Computational Linguistics*, 38(1):173–218.

Michael Lebowitz. 1985. *Story-telling as planning and learning*. Poetics, Volume 14.

Matt Madden. 2006. *99 ways to tell a story*. Jonathan Cape, London.

James R. Meehan. 1997. *Tale-Spin, an interactive program that writes stories*. The Fifth International Joint Conference on Artificial Intelligence at MIT, Cambridge, Massachusetts.

Nick Montfort. 2007. *Generating Narrative Variation in Interactive Fiction*. Ph.D. thesis, University of Pennsylvania.

Nick Montfort. 2009. Curveship: An interactive fiction system for interactive narrating. In *Proceedings of the Workshop on Computational Approaches to Linguistic Creativity*, pages 55–62, Boulder, Colorado. Association for Computational Linguistics.

Nick Montfort and contributors. 2021. Curveship-js github repository. [Online; accessed 5-December-2021].

Nick Montfort, Rafael Pérez y Pérez, D. Fox Harrell, and Andrew Campana. 2013. Slant: A blackboard system to generate plot, figuration, and narrative discourse aspects of stories. In *Proceedings of the International Conference on Computational Creativity (ICCC) 2013*, pages 168–175.

Gerald Prince. 1982. *Narratology: The form and functioning of narrative*. Walter de Gruyter, Berlin.

Rafael Pérez y Pérez. 1999. *MEXICA: A Computer Model of Creativity in Writing*. Ph.D. thesis, The University of Sussex.

Raymond Queneau. 1981. *Exercises in style*. New Directions, New York.

Emily Reif, Daphne Ippolito, Ann Yuan, Andy Coenen, Chris Callison-Burch, and Jason Wei. 2021. A recipe for arbitrary text style transfer with large language models. *arXiv preprint arXiv:2109.03910*.

Mark O. Riedl and Michael R. Young. 2010. *Narrative Planning: Balancing Plot and Character*. Journal of Artificial Intelligence Research, Volume 39.

Shlomith Rimmon-Kenan. 1983. *Narrative ficiton: Contemporary poetics*. Methuen, London.

Marie-Laure Ryan, James Ruppert, and John W. Bernet. 2004. *Narrative across media: The languages of storytelling*. University of Nebraska Press, Lincoln, Nebraska.

Laurence Sterne. 1986. *A sentimental journey through France and Italy*. Harmondsworth; Penguin Books, Middlesex, England; New York.

Scott R. Turner. 1994. *The Creative Process: A computer model of storytelling and creativity*. Lawrence Erlbaum Associates, Hillsdale, New Jersey.

Shufan Wang and Mohit Iyyer. 2019. Casting Light on Invisible Cities: Computationally Engaging with Literary Criticism. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1291–1297, Minneapolis, Minnesota. Association for Computational Linguistics.

Terry Winograd. 1972. Understanding natural language. *Cognitive Psychology*, 3.

Sina Zarrieß and David Schlangen. 2016. Easy things first: Installments improve referring expression generation for objects in photographs. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 610–620, Berlin, Germany. Association for Computational Linguistics.

The concept of nation in nineteenth-century Greek prose fiction through computational literary analysis

Fotini Koidaki
Department of Modern Greek
and Comparative Studies
Aristotle University of Thessaloniki
54124 Thessaloniki – Greece
coidacis@gmail.com

Despina Christou
Department of Informatics
Aristotle University of Thessaloniki
54124 Thessaloniki – Greece
christoud@csd.auth.gr

Aikateriki Tiktopoulou
Department of Modern Greek
and Comparative Studies
Aristotle University of Thessaloniki
54124 Thessaloniki – Greece
atiktopo@lit.auth.gr

Grigorios Tsoumakas
Department of Informatics
Aristotle University of Thessaloniki
54124 Thessaloniki – Greece
greg@csd.auth.gr

Abstract

How can the construction of national consciousness be captured in the literary production of a whole century? What can the macro-analysis of the 19th-century prose fiction reveal about the formation of the concept of the nation-state of Greece? How could the concept of nationality be detected in literary writing and then interpreted? These are the questions addressed by the research that is conducted in this paper and which focuses on exploring how the concept of the nation is figured and shaped in 19th-century Greek prose fiction. We propose a methodological approach that combines well-known text mining techniques with computational close reading methods in order to retrieve the nation related passages and to analyse them linguistically and semantically. The main objective of the paper at hand is to map the frequency and the phraseology of the nation-related references, as well as to explore phrase patterns in relation to topic modeling results.

1 Introduction

Even though Literature is not the accurate representation of reality, the literary text is actively participating in the formation of the cultural identity of the community in which it is written and read. As far as the national identity is concerned, the literary discourse is interwoven with the processes of formation and negotiation of the nationalistic ties of a community, since it not only is embodied through the most important nation-building tool, Language, but reflects and, at the same time, shapes the collective conscious. The paper at hand presents and discusses the data-set, as well as the results, of a research project that was focused on exploring and analysing how the concept of nation is figured in Greek prose fiction of the 19th century.

The 19th century is of considerable cultural and political importance for Greece since this was a decisive period for the flourishing of Greek letters and for the course of Greek history in general. The 19th century was the century of Greek ethnogenesis, which had a great influence on the European movement of nation-states (Beaton, 2021). The Greek revolution of 1821 was the first successful liberal-national uprising in Europe because, after the fall of Napoleon and the Congress of Vienna, the authoritarian monarchies had been restored in the continent. The Greek War of Independence that launched exactly 200 years ago, ended the 400 years of Ottoman rule in the Greek territory, and resulted in the establishment of the first Hellenic Republic that led to the formation of the modern state of Greece. During the 19th century and in parallel with the building of independent Greece, typography emerged and flourished in the newly-born state. Newspapers, journals, and publishing houses appeared and multiplied over the century, revealing this way the escalation of publishing activity in Greece and the consequent flowering of

Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH), pages 75–84

letters (Pouliasis, 2021). The explosion of literary publishing, which scaled up especially around 1880-1900, as well as the increase printing industry is evidence of the rise of literacy levels among Greek citizens. At the same time, the role of literary people in the Greek Language Controversy that arose before the Revolution, the refuge that several revolutionists found in literature and particularly in self-narration, as well as the connection established between national and literary language, provide evidence demonstrating the service of literature to the national purpose.

In this context, our research aims to explore how the nation concept is figured in the 19th-century Greek prose fiction in order to address questions regarding the presence of the concept of nation in novelistic tradition, but also the discussion and the conceptualisation of it. The rest of this paper is structured as follows. Section 2 presents background material concerning computational analysis of the notion of nation in literature. Next, we outline the dataset generated (Section 3), the methodology developed (Section 4), as well as the findings of our research (Section 5). Finally, Section 6 presents the conclusions of our work.

2 Background

The 19th century is the century of the formation of nation states. The relation between literature and (romantic) nationalism has been discussed for almost a century (Bradsher, 1921; Giffin, 1945; Fisher, 1980). Literature is known to be one of the major contributory factors in the nation-building process, both because it is embodied through Language -that is one of the most important characteristics of collective identity- and because it has the power of building a national audience. In the case of the creation of the Greek nation state, researchers from different fields have focused on aspects of the national character's building in the pre-revolutionary and post-revolutionary years and have commented on the reflection of these aspects in the periodical press of the 19th century, in texts of various types written by Greek and European scholars during this period, in specific authors and in selectively chosen literary texts.[1] The investigation of the issues of nation-building and national self-determination in a corpus of modern Greek prose of the 19th century is attempted for the first time here.

On the other hand, the attempts to computationally analyse the representation of nationality in literature are limited. For example, Weenink (2018) explores the national trends in Gothic fiction using topic modeling along with a qualitative approach of close reading and contextualization, while Erlin et al. (2021) count the geographic mentions (using NER models, part-of-speech tagging, place names gazetteer) and national phrases (taken form the "history of" Wikipedia page) in narration to find out the level of national attention in contemporary European fiction across multiple national contexts. To our knowledge, we are the first to computationally analyse the concept of nation in modern Greek fiction of the 19th century.

3 Data

To explore the concept of the nation in 19th century Greek prose fiction, we needed to create a literary corpus because, as yet, there are no corpora of 19th century Greek prose fiction available. Considering the project timescales and the financial factors, we built a micro-corpus, which we digitised and further processed to develop a final dataset that could provide sufficient knowledge to answer the questions posed by this project.

Among the corpus-building criteria that we defined, and which were related to the language, the date, and the type of publication, the comprehensive and balanced representation of the 19th-century Greek novelists was very important. Therefore, we didn't limit our corpus to the literary canon. Instead, we decided to include every novelist published in the Greek language during the 19th century, such as lesser-known artists, novelists publishing in districts outside the capital of Athens, a foreign author who wrote in the Greek language and printed outside Greece, as well as an unknown author. However, because the literary production of each author differs in size, we decided to select one title of each novelist to save time and maintain balance. Consequently, we have chosen the most representative work of each author, while taking into account the publication dates to ensure enough samples from every decade of the century. Finally,

[1] From the rich bibliography, we mention here the volume edited by Roderick Beaton and David Ricks Beaton and Ricks (2009), the monographs by D. Tziovas Tziovas (2017) and A. Politis Politis (2017) and the two volumes of the conference proceedings entitled Greekness and Diversity? Cultural Mediation and "National Character" in Nineteenth Century, edited by A. Tampaki, and Ou. Polikandrioti Tampaki and Polikandrioti (2016).

we built-up a catalog of 91 novels, corresponding to 90 novelists and an anonymous title. The 91 novels are all written in the Greek language and published inside and outside Greece or Greek-speaking territory between 1811 and 1901.

The final corpus, which corresponds to more than 16000 pages, was digitised on the Transkribus[2] platform with an OCR model that we trained by transcribing a 150-page (35.223 words) sample of the three most common fonts of our corpus (Didot, Porson, Teubner). Table 1 shows our OCR model character errors.

Metric	Train	Validation
CER	0.62%	1.44%

Table 1: OCR's Character Error Rate (CER) on train and validation sets

To reduce the noise that would affect the accuracy of the text mining tasks, we massively processed the OCRed files by treating cases such as hyphenation, watermarks, repetitive page headers, additional noisy characters, etc. Figure 1 demonstrates word-counts distribution in the 19th century after the post-processing phase.

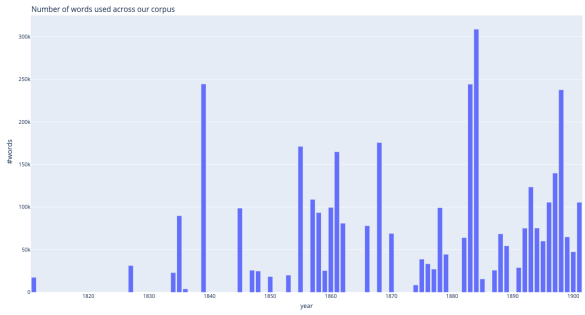

Figure 1: Word counts distribution in the 19th century

As we focus on such a specific topic as the nation is, we decided to prioritise a keyword-based passage retrieval approach, since this would allow us to collect and analyse all the nation-related passages contained in our corpus. The main idea was to establish a knowledge base featuring all the direct references to the concept of the nation comprised in Greek prose fiction, and which could be further studied, enriched, and analysed to answer the questions our project addresses. Therefore, the first step was to design the keyword query by defining the keywords of interest. Considering the available Greek dictionaries of the 19th century, we defined seven keyword-stems related either to the nation in general or to the Greek nation in particular (translated from Greek language: nation, homeland, tribe, genus, Hellas, Greek, Romios [3]). For each of these keywords, we built a vocabulary, where we specified all the different grammatical forms derived from the seven stems, their possible spelling variations, as well as the unacceptable lexical types that may also derive from the same stems. These vocabularies allowed us to capture as many tokens as possible regardless of those including typos, spelling mistakes, idiom variations, or OCR noise.

The retrieved passages were organised on a tabular dataset and further enriched with various meta-information related to the passage-itself, the related keyword, and its parent document. [4] The passage retrieval was defined at the sentence level and not at the paragraph or phrase level, as the sentence, which is the smallest grammatical unit expressing a complete thought, would provide us with a sufficient sample to observe the concept of the nation in its context. Therefore, for each keyword captured, the query returned the complete sentence containing it alongside the matched keyword-stem and other useful meta-information. This suggests that in cases where more than one keyword was located in the same sentence, as many entries were created as the keywords captured in the sentence. Therefore, to monitor the findings' density and distribution in the documents, we assigned numeric identifiers to the sentences ("sentence-id") to observe the references through sentences. In the first step, a typical entry comprised the matched keyword, the complete grammatical sentence containing it, the "sentence-id", as well as useful metadata about the parent document (title, author, publication date, total sentence count, total words count).

However, the dataset was evaluated and further enriched. We repeatedly evaluated at least a random 10 percent of every keywords' results in order to fine-tune and optimise the keyword-vocabularies of the query. Based on the deficiencies observed, we decided to further evaluate the findings of two specific keywords (tribe, genus) in order to reduce noise caused by the findings that were not related to the concept of the nation. Having denoised the results as much as possible, we semantically enriched further the dataset with the geographical entities

[2] https://readcoop.eu/transkribus/

[3] The Orthodox Christian Greek-speaking citizens of Ottoman empire

[4] Available upon acceptance.

that we automatically extracted from the retrieved passages. The final, evaluated and enriched version of the dataset is comprised of 8015 entries capturing and annotating individual references to the concept of the nation in the corpus of 19th-century Greek prose fiction.

4 Methodology

The generated dataset forms a topic-specific knowledge base that brings together all the nation-related references captured in the ninety-one nineteenth-century Greek novels presented above. To address the questions posed by the research, we subjected the dataset both to a statistical and linguistic analysis by combining well-known text mining techniques.

For instance, by grouping and filtering the dataset based on the collected metadata, we managed to develop a successively better picture about the frequency of the nation-concept occurrence and its presence in the corpus of the 19th century in general.

In the following, we deepened into a semantic analysis of the nation-related references to further understand the context describing the concept of nation. Precisely, we extracted the collocations around each nation-related keyword and all keywords to better understand the discussed topics in each keyword and their entire set, respectively. The lack of language processing tools (pos tags, stopwords) for the specific Greek language forms (Dimotiki and Katharevousa) of the 19th century - also characterizing our corpus - led us to combine tools developed for ancient- and modern- Greek language. For instance, the lack of pos taggers for the specific era guided us to enhance the stopword list by combining ancient- and modern-Greek stopwords lists, which we also enhanced with common OCR misspellings.

At last, we extracted clusters of words, i.e., words that frequently occur together, from the passages using topic modeling. Precisely, we used the Latent Dirichlet Allocation[5] (LDA) (Blei et al., 2003), the prevailing generative probabilistic topic model, to detect the thematic information of our archive. After hyper-parameters tuning, we concluded in fourteen (14) topics, among which there exists a topic that seems to be highly related to the concept of the nation-state. Finally, in order to read and further deepen our analysis on the topic modeling results, we exploited the knowledge gained from all the above-mentioned data-mining techniques.

[5]We used the Gensim (Rehurek and Sojka, 2011) python library

5 Analysis

5.1 The presence of the concept of nation in the corpus

Dataset sorting, grouping and filtering provided interesting insights into the concept of nation in the 19th century Greek prose fiction corpus. The aggregation of data based on variables such as "keyword", "date" or "author" provided different perspectives on the frequencies of the nation-related references retrieved from 19th-century Greek fiction.

In particular, when we attempted to measure the occurrence of the concept of the nation in each document, we noticed that the keywords' occurrences were not always directly proportional to the length of each book. In fact, we compared the total number of sentences containing nation-related references (see in Figure 2 the red bars with y axis to the right) to the total length of each document measured in sentences (see in Figure 2 the cyan bars with y axis to the left). A sentence, of course, may have contained more than one reference to the nation. However, each sentence containing nation-related references was conceived as a single instance of a complete reference to the concept of the nation. We used the sentence as a measurement unit to create measurable quantities and properly map the complete references to the concept of the nation in our corpus. For example, Figure 2 shows that some shorter documents contain more references than longer ones, thus suggesting that the document's length should also be taken into account in our effort to quantify the presence of the concept of nation in the corpus.

The "sentence-id" was the variable that allowed us to measure our findings at the sentence level. Using the aforementioned measurements we estimated the rate of nation-related sentences for each document to observe the extent to which each novel refers to the concept or even the topic of the nation. However, when we generated a timeline graph of the reference rates (see Figure 4) we got a completely different view compared to that of the total number of the tokens retrieved (see Figure 3). More precisely Figure 3 presents the total number of tokens per publication date, while Figure 4 shows

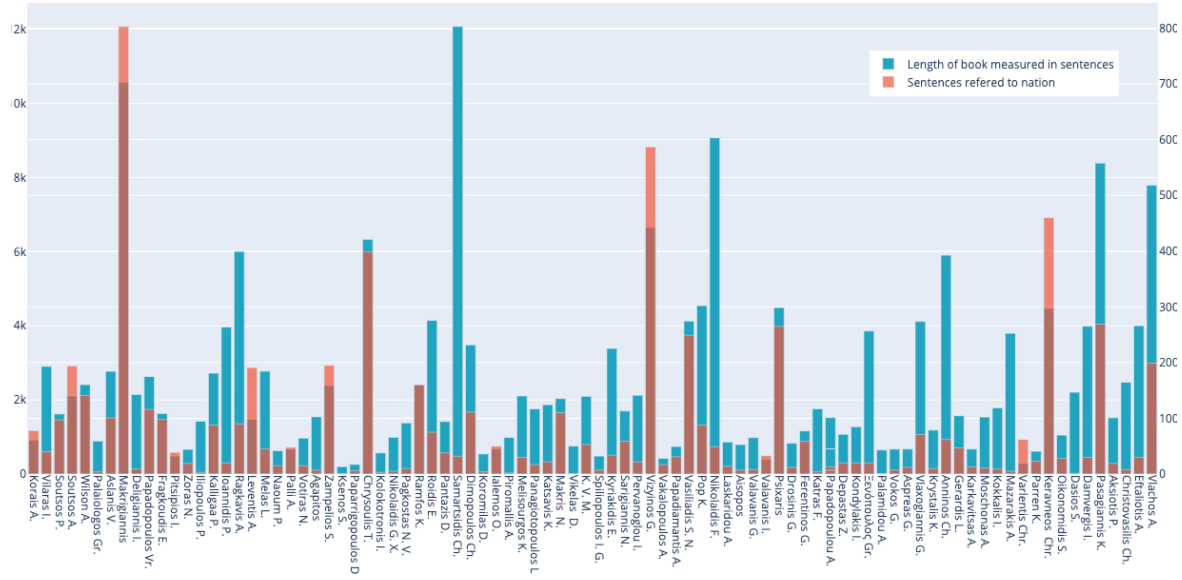

Figure 2: The sentences referred to nation and the total length of each book in sentences.

the reference rates per publication date. Figure 4 displays the discussion about the concept of nation in the corpus being more active and intense during the last two decades of the century. Figure 4, which takes into account the length of the sources, offers a more complete and objective look into the presence of the concept of the nation in the corpus we built.

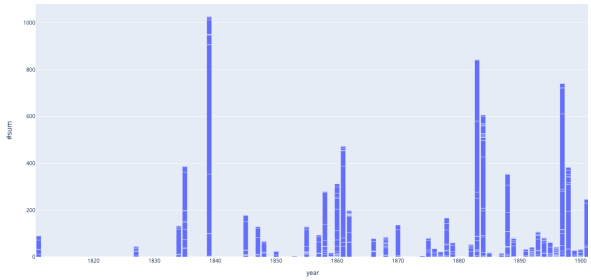

Figure 3: The nation-related tokens in a timeline

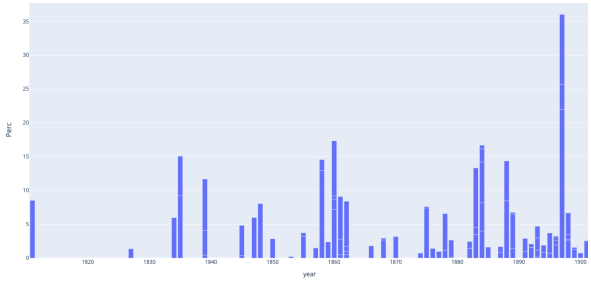

Figure 4: The rate of nation-related sentences in a timeline

5.2 The presence of the keywords in the corpus

By elaborating further on the retrieved data, we could obtain information about the presence and frequency of each defined keyword. We decided to count the number of tokens retrieved by each keyword to get a better insight into the presence of the concept of nation in the body of 19th-century Greek fiction. This enabled us to observe the frequency of the defined keywords in the corpus across the 19th century. As we can see in the line chart of Figure 5, keywords such as “Hellas”, “homeland” are by far most frequent, unlike keywords such as “tribe”, “genus”, “Romios”[6] and “Graecus”/“Graecòs”[7] which are less frequently used in our corpus. It is very interesting that the national adjectives “Romios” and “Graecus” show very low numbers of occurrences and disappear at certain periods. For instance, the national adjective “Romios” shows the smallest frequency of occurrences among the keywords retrieved and it does not appear before 1839, as it is absent from the first seven documents of the corpus. More interestingly,

[6] “Romios” or “Romaios” was called the Orthodox Christian citizen under the Ottoman Empire. From the beginning of the 19th century, the term “Romios/Romaios” becomes equivalent to the term “Graecus”/“Graecòs” and declare the social and national identity of the Greek.

[7] The term “Graecus”/“Graecòs” which, from the Renaissance onwards retains the wider meaning of the speaker of the Greek language, during the 18th century declares the inhabitant of southeastern Europe who speaks Greek, and from the beginning of the 19th century is identified with the Greek ethnicity and means the Greek.

the national adjective "Graecus" does not appear in our corpus after the year 1888, as it is completely absent from 31 documents published between 1889 and 1901. The absence of the national adjective "Graecus" from 1888 onwards could be related to the general tendency observed in non-literary texts to avoid both "Graecus" and "Romios" and to use "Ellin" widely (Katsiardi-Hering, 2018).

5.3 The context of the concept of the nation

Based on the above-mentioned statistic analysis of the frequency and the dispersion of the concept of nation in our corpus, we continued by maintaining a top-down approach since we next focused on the context of the retrieved tokens. The main objective here was to obtain a more detailed picture of how the concept of nation was rendered and expressed in the 19th-century literary corpus we built. What was the vocabulary with which the concept of the nation is related and defined? What other concepts was it associated with?

Since there are no Part-of-Speech (POS) taggers for 19th-century Greek language, which would provide us with information about the syntactically related words to the tokens retrieved, we performed a collocation analysis. Collocation analysis revealed the context in which the concept of nation is shaped and defined in our corpus. The collected collocations were visualised with word clouds (also known as a tag cloud) which offered us a practical and compact insight into the context of each keyword individually and all the keywords as a whole. For instance, the 40 words that collocate most frequently with the keyword "nation" are shown in Figure 6.

As can be seen in Figure 6, of all the keywords we defined, only "Greece" (including the national adjective Greeks) and "homeland" appear to be associated with the keyword of the "nation". In fact, the token "Greece" is the most common collocation of the word "nation", and this indicates that the references to the term "nation" in our corpus refer primarily to the Greek nation. Except "Greece", the other geographical name that we can see on the cloud is the word "Europe" (upper right corner), which is an evidence of the strong political and ideological dimension Europe had for the construction of the modern state of Greece. However, some of the fundamental concepts of the Greek national integration seem to be condensed here in the high-frequency collocations of the keyword "nation", that include words such as "Greece","today", "justice", "fathers", "homeland", "freedom". Therefore, the issue of nation-building appears here as the urgent and topical demand for the restoration of justice and freedom. Furthermore, "fathers" that is a common Greek metaphor for the ancestors, is a central component of the national identity as it links together the past with the "today" and the "future" of the nation.

On the other hand, Figure 7 summarises the 40 words found in our corpus to collocate most frequently with all the seven keywords we defined ("nation", "Greece", "homeland", "genus", "tribe", "Graecus", "Romios"). The word cloud depicted in this Figure does not include the keyword-stem tokens. Even though it was usual for a keyword to be in a common context with other keywords (ie. in Figure 6 we saw "nation" to collocate with "Greece"), we use here a keyword-free cloud in order to observe their common context without any interference.

Interestingly enough, one of the high-frequency tokens of the word cloud of Figure 7 is the word "children", which is a metaphor for the citizens, since in phrases like "children of Greece" the to mean the people of Greece. In view of that, we can observe that this cloud reveals concepts that together emphasise the struggles for national independence and collectivisation, which is summarised in words like "today", "life", "blood", "land", "people", "freedom", "Union", "we" and "others". At the same time, the high-frequency tokens of the cloud are concepts like "today", populace ("children" and "people"), "blood" "life" and "name", which reflect the existential importance of the nation-building process. The notions of today and the populace could once again indicate how current and urgent the issue of nation-building was, while the blood, life, war, and guns could highlight the importance of the issue posed in between life and death. What is at stake in the struggle for national Independence is life itself. The token "name" is directly connected with the national designation, whether it refers to the collective name of the people or the personal name of a character.

Furthermore, both word clouds include concepts that are related to components of the national identity and which are dominant in shaping the national and collective consciousness. Concepts like these are freedom, land, language, ancestors (i.e. fathers), religion and God, and especially as regards Greece, honor, soul, glory and heart.

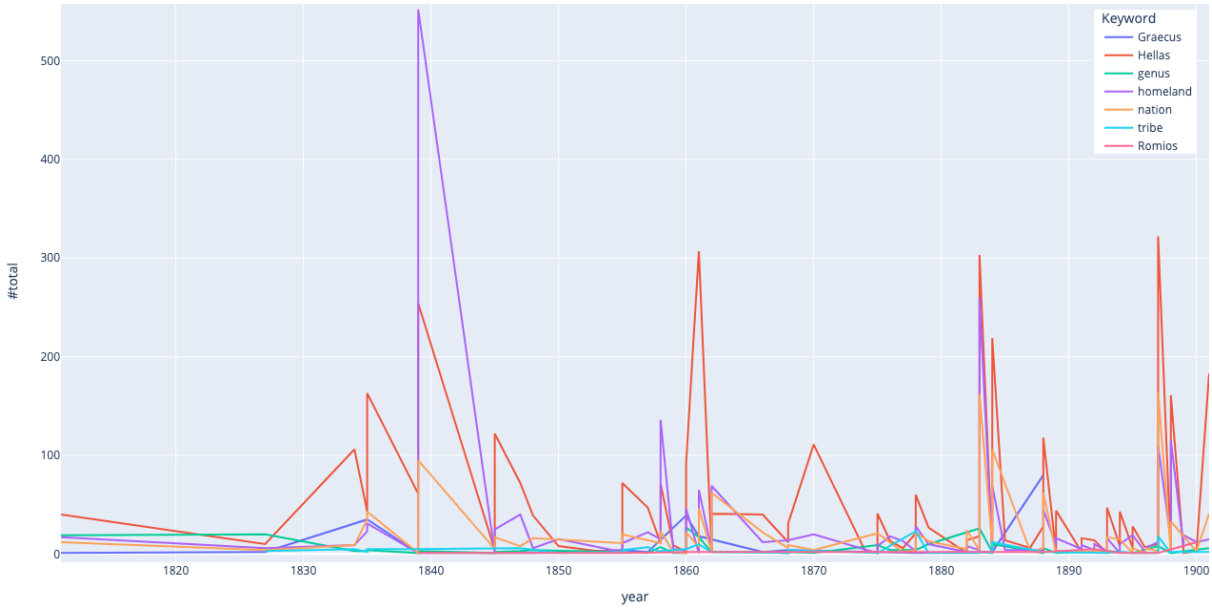

Figure 5: The keywords frequencies on a timeline.

Figure 6: Collocation of "nation" keyword.

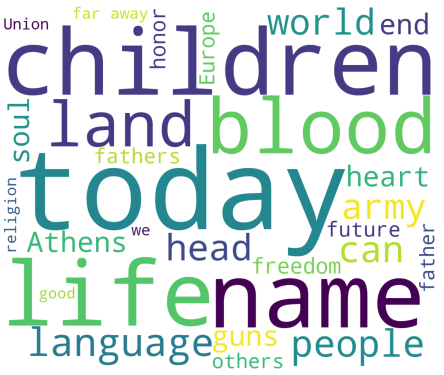

Figure 7: Collocation of all the seven keywords (nation, Greece, homeland, genus, tribe, Graecus, Romios).

5.4 The topic of nation-state

The unsupervised topic modeling analysis allowed us to automatically discover the thematic structures found within the entire corpus of the 19th-century Greek prose fiction. Even though this analysis was useful for the initial exploration of the corpus, we further exploited the topic modeling result by identifying the link between the nation theme and the extracted topic contents.

Out of the fourteen topics we extracted, we can see in Figure 8 that only the four most prevailing formed distinctive clusters. On the other hand, the ten least prevailing topics formed overlapping clusters, each of which was observed to contain heroes' names and was therefore associated with an individual novel. Likewise, the fourth topic, found in the right bottom corner of the figure, was observed to include vocabulary referring to the plot and the characters of a particular novel, which has a unique theme, different from the rest of the corpus. However, the three most distinctive topics were associated with more general literary worlds, recurrent in our corpus. The first was comprised of terms related to the family relationships as well as to the religion (terms: god, head, heart, husband, daughter, blood, father, name, etc). The second topic included terms referring to the countryside, youth, and education (terms: home, child, village, language, brother, teacher, nature, etc). Finally, the third topic (in red circle) was observed to be highly associated with the concept of nation, since, as discussed in further detail below, it included terms referring to the State of Greece and its 19th-century history.

The third topic is considered to be the only one

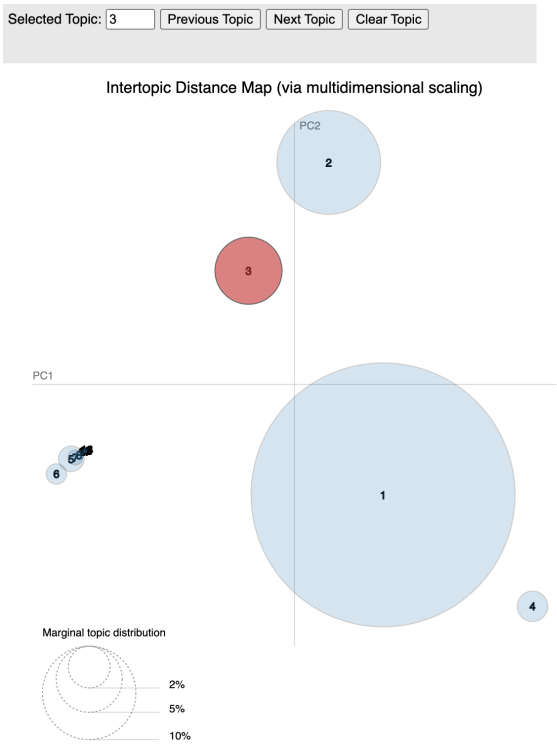

Figure 8: The topic which is referred to the nation-state of Greece.

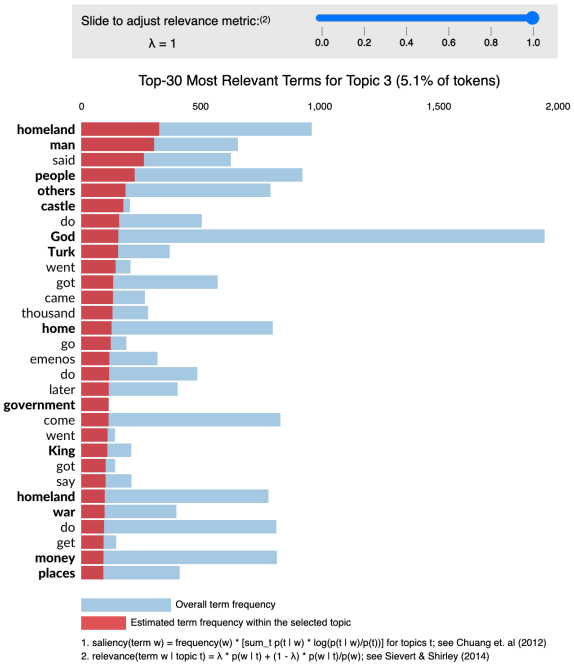

Figure 9: The topic which is referred to the nation-state of Greece.

that is directly and almost completely related to the concept of nation 9). In fact, the group of words that consists the cluster of the third topic seems to compose a condensed narration of 19th-century history of the newborn nation-state of Greece. For this topic, the most relevant term (given $\lambda = 1$) is the noun "homeland" which, at the same time, is the highest frequency keyword among those that we have defined as suggestive of the concept of the nation. Out of the thirteen nouns constituting the topic, the nine are terms already found in the collocation analysis of the nation keywords, such as "homeland", "man", "people", "others", "God", "Turk", "King", "land", and "war". The verbs that appear among the nouns constitute common enough verbs such as "do", "say", "come", "go", or "get" indicating the action of the nation-building progress.

By reducing the relevance value below 0.5, we observe the formation of word groups that refer to either the government establishment or the liberal-national revolution. For example, by setting the value of λ at 0.4 we get a word group such as "caste", "government", "people", "governor", and "homeland that seems to reflect the issue of governance in the newborn state of Greece. While, at 0.25 (and below that) the words "ammunition", "killed", "people", "Kolokotronis", "warriors", "fought", "patriots", "drachma" (coin currency), and "elections" are added at the bottom. Both word groups together may form a direct reference to the issue of the warriors' demand to participate in the governance of the free state of Greece. Considering the different topic contents produced by the multiple relevance adjustments, we may say that the third topic is directly connected with the discourse about the establishment of the first Hellenic Republic and the formation of the modern state of Greece. Even though the topic contains traces of the nation theme, it is clear that it also extends to statehood issues like governance or economy.

6 Conclusions

Considering the challenge of the 19th-century Greek language, computational analysis of prose fiction offered us valuable insights into the special issue of the construction of the national identity. Using the keyword-based retrieval technique we managed to track down all the direct references to the concept of the nation. The statistical analysis of these findings allowed us to observe the size and the frequency of the nation-relate discussion across our corpus and its timeline. In order to produce accurate and realistic measurements, we used metadata about the length of each document that

enabled us to estimate the rates of nation-related references across the corpus. As an overall observation on the presence of the concept of nation in our corpus, it was noticed an increasing and escalating tendency during the last two decades of the century.

Focusing on the phraseology of the nation-related references, we also attempted to identify the vocabulary through which the concept of the nation is expressed and shaped. By counting the retrieved keywords, we confirmed the Greek peoples' preference -witnessed by non-literary sources- to self-determine themselves as *Hellines* (singular *Hellin*) rather than *Romios* or *Graecus*. Moreover, the keywords we defined as direct references to the nation, were further analyzed with a focus on their collocations. The collocations extracted reflect aspects of the Greek national identity and milestones of the Greek history of the 19th century related either to the War (of Independence) or the challenges of the nation-building process.

However, the collocations' word clouds may provide a slightly different picture of that we get by analysing the topic modeling results. In fact, the results of the collocation analysis were found to be associated with one of the thematic patterns that have the highest probability in the corpus. This topic, which is the third words' cluster in the topic modeling analysis, brings together terms that are linked to the nation and refer, as mentioned above, either to the War of Independence or to the institutionalisation and the governance. Considering the several versions that the topic may form by the adjustments of the relevance value, we notice that the concept of the nation is present in a topic that, at the same time, reflects the pragmatic challenges of the state-building process (i.e. government, governor, king, money, elections). At the same time, the collocations bring together terms that make up the picture of romantic nationalism, since they combine concepts like justice, idea, freedom, glory, honor, soul, heart, life, hope, belief, Union.

The methodological approach described above enabled us to observe the concept of the nation in the corpus of 19th-century Greek prose fiction we built. However, the result, as well as the method, could be further enriched and developed both by extending the content of the corpus and by developing a 19th-century specific language toolkit, which would provide us with more detailed and comprehensive insights into the 19th century Greek Literature.

Acknowledgments

The research presented in this paper was supported by the project "Semantic analysis of 19th-century modern Greek fiction with text mining techniques" (EDBM 2014-2020)" which has been funded by the European Union and Greek national funds through the program "Supporting researchers with an emphasis on young researchers" (call code EDBM34), operational program "Development of Human Resources, Education and Lifelong Learning", NSRF 2014-2020.

We would also like to thank the anonymous reviewers for their careful reading of our paper and their many insightful comments and suggestions.

References

Roderick Beaton. 2021. The significance of the 1821 revolution for greece and the world. *ekathimerini-com*.

Roderick. Beaton and David. Ricks. 2009. *The Making of Modern Greece : Nationalism, Romanticism, and the Uses of the Past (1797-1896)*. Taylor Francis Ltd.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022.

Earl L. Bradsher. 1921. Nationalism in our literature. *The North American Review*, 213(782):109–118.

Matt Erlin, Andrew Piper, Douglas Knox, Stephen Pentecost, Michaela Drouillard, Brian Powell, and Cienna Townson. 2021. Cultural capitals: Modeling 'minor'european literature. *Journal of Cultural Analytics*, 1(2):21182.

John H. Fisher. 1980. Literary culture: Nationalism and the study of literature. *The American Scholar*, 49(1):105–110.

Mary E. Giffin. 1945. Nationalism and english literature. *College English*, 6(6):310–313.

Olga Katsiardi-Hering. 2018. Ellin, graikos, romios: from multinational to national. In *Ellin, Romios, Graikos: Collective Identitifications and Identities*, volume 7, chapter 1, pages 19–35. Evrasia, Athens.

Alexis Politis. 2017. *Romantic Literature in the National State*. Philologia. Crete University Press, Herakleion.

Efstathios Pouliasis. 2021. *The perception of the Greek Revolution in the Public Sphere of the GreekState (1832-1920). Dissertation*. Ionian University: Faculty of History Translation, Corfu.

Radim Rehurek and Petr Sojka. 2011. Gensim–python framework for vector space modelling. *NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic*, 3(2).

Anna Tampaki and Ourania Polikandrioti. 2016. *Greekness and Diversity; Cultural Mediation and National Character in Nineteenth Century*, volume A B. Athens: National and Kapodestrian University of Athens National Research Institute, Athens.

Dimitris Tziovas. 2017. *The Cultural Poetics of Greek Fiction: From Interpretation to Ethics*. Philologia. Crete University Press, Herakleion.

Maartje Weenink. 2018. *Who's Afraid of the Big Bad?: The Representation of Nationality in "British" Gothic Fiction 1750-1840. A Computational Approach to Topics in Fiction. Thesis*. Radboud University: Department of Historical, Literary and Cultural Studies, Nijmegen, Netherlands.

Logical Layout Analysis Applied to Historical Newspapers

Nicolas Gutehrlé
Centre de Recherches Interdisciplinaires
et Transculturelles (CRIT),
Université de Bourgogne Franche-Comté
30 rue Mégevand, 25000 Besançon, France
nicolas.gutehrle@univ-fcomte.fr

Iana Atanassova
Centre de Recherches Interdisciplinaires
et Transculturelles (CRIT),
Université de Bourgogne Franche-Comté
Institut Universitaire de France (IUF)
30 rue Mégevand, 25000 Besançon, France
iana.atanassova@univ-fcomte.fr

Abstract

In recent years, libraries and archives led important digitisation campaigns that opened the access to vast collections of historical documents. While such documents are often available as XML ALTO documents, they lack information about their logical structure. In this paper, we address the problem of logical layout analysis applied to historical documents. We propose a method which is based on the study of a dataset in order to identify rules that assign logical labels to both block and lines of text from XML ALTO documents. Our dataset contains newspapers in French, published in the first half of the 20th century. The evaluation shows that our methodology performs well for the identification of first lines of paragraphs and text lines, with F1 above 0.9. The identification of titles obtains an F1 of 0.64. This method can be applied to preprocess XML ALTO documents in preparation for downstream tasks, and also to annotate large-scale datasets to train machine learning and deep learning algorithms.

1 Introduction

One important challenge in digital humanities is the efficient exploitation and processing of scanned textual documents (archives, documentary funds, ...). For example, historical documents such as newspaper archives are prime resources for historians (Tibbo, 2007). Thanks to the important digitisation campaigns led by libraries and archives, vast collections of historical documents have been made easily accessible. However, the majority of these documents are available only as scanned images (e.g. in PDF format) which makes them difficult to explore in a text processing perspective. Extracting the text content from such documents requires at least the following three steps: Optical Character Recognition (OCR), physical layout analysis (PLA) and logical layout analysis (LLA).

Physical layout analysis (PLA), which is also sometimes called *document layout analysis*, consists in identifying physical regions of the document, with their text content and boundaries. Such regions can correspond to sections and lines of text, but also to figures, tables, etc. PLA also defines the reading order of the document, which corresponds to the linear order in which the different regions appear. This is particularly important for documents that have multi-column layouts. One commonly used output format of PLA is the XML ALTO format[1]. *Logical layout analysis (LLA)*, sometimes called *logical structure derivation* and *structure understanding*, consists in identifying the document structure elements and their categories i.e. title, header, paragraph, table, etc. Such logical elements can integrate one or more regions in the document that have been identified by PLA.

Physical and logical layout analyses are necessary steps in the processing of documents for a large number of applications, including Information Retrieval, information extraction, Table of Content extraction, text syntheses, and more broadly document understanding.

In this article we focus on the problem of logical layout analysis (LLA). We describe a methodology for logical layout analysis, where logical labels are assigned to physical layout entities. The input of our processing pipeline is the physical layout analysis of documents in the XML ALTO format.

The rest of the article is organised as follows: the following section presents the related work on logical layout analysis. Section 3 present our train and test datasets. Section 4 presents the methodology that we propose and section 5 proposes an evaluation of the implemented processing pipeline. Finally, we propose a conclusion and a discussion.

[1] ALTO: Technical metadata for layout and text objects: https://www.loc.gov/standards/alto/

Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH), pages 85–94
December 19, 2021.

2 Related works

An important body of research around physical layout analysis of printed documents has been produced in the end of the XXth century. Several algorithms have been proposed such as the X-Y Cut algorithm (Nagy et al., 1992), the Docstrum algorithm (O'Gorman, 1993) or the Voronoi diagram based algorithm (Kise et al., 1999). Furthermore, the processing of handwritten documents requires specific techniques, such as the "droplet" technique to identify text line by Bulacu et al. (2007), or neural networks as in Chen and Seuret (2017), where each pixel is labelled as text or not.

Existing logical layout analysis systems make use of various methods that go from heuristic systems to more recent architectures using neural networks. Some heuristic systems use grammars such as stochastic or attributed grammars, where the document is represented as a string of symbols, e.g. Namboodiri and Jain (2007). In their work, the grammar describes multiple production rules, each associated with a logical label. The string of symbols is then parsed by the grammar in order to extract logical labels. Other systems, such as LA-PDFText (Ramakrishnan et al., 2012) or DeLoS (Niyogi and Srihari, 1995), use rules that state the condition a physical block must meet to be given a logical label. For instance, DeLoS system uses first-order predicates in order to infer the logical category of a physical block.

While heuristic systems provide good results, they are often dedicated to specific layouts, and need to be adapted to work on other layouts. To tackle this problem, Klampfl and Kern (2013) created a system for logical layout analysis on scientific articles in PDF format that combines heuristic rules with unsupervised-learning models such as k-means or Hierarchical Agglomerative Clustering (HAC). This system is made up of several detectors, each learning geometrical and textual features from the document in order to identify a specific logical label. Some rules using text occurrences are also used to help the model, such as finding the keywords "Table" or "Fig." to identify table or figure blocks.

More recent works use neural networks for logical layout analysis. As noted by Akl et al. (2019), CNN or LSTM architectures work better than classical neural networks because of the sequential nature of the documents. This task also benefits from the use of word-embeddings such as fastText, Flair or GloVe which give a better encoding of textual data than simple one-hot encodings, as in Zulfiqar et al. (2019). Neural network systems can be trained on big datasets such as the Publaynet dataset (Zhong et al., 2019) or the Medical Articles Record Groundtruth (MARG) for physical and logical layout analysis purposes.

Considering the task of processing historical documents, several small datasets exist such as the DIVA-HISDB dataset (Simistira et al., 2016) which contains 150 annotated pages of three different medieval manuscripts or the European Newspapers Project Dataset (Clausner et al., 2015) which contains 528 documents. Other datasets in non-European languages exist, such as the PHIBD dataset (Hossein Ziaie Nafchi and Cheriet, 2012), which contains images of 15 Persian historical and old manuscript, and the HJDataset by Shen et al. (2020), which contains 2271 Japanese newspapers published in 1953, which was generated in a semi-automatic way. All of these datasets are too small to be used for machine learning or neural network approaches.

Hébert et al. (2014) deal with the task of article segmentation by a Conditional Random Field (CRF) model with heuristic rules to perform logical analysis. First the CRF model labels pixel as titles, text lines, or horizontal and vertical separators, then heuristics rules describing usual article layouts are applied to that classification. In both cases, bad results were caused by the quality of the scan or the quality of the OCR output. On the other hand, Riedl et al. (2019) deal with article segmentation by looking at the similarity between segments of texts. These segments are computed either by using the Jaccard coefficient and their word distribution or by computing the cosine similarity between word-embeddings. The similarity between blocks is then computed using the TextTiling algorithm (Hearst, 1997)

Most common approaches to LLA are not suited for historical documents because the document layout changes over time. For example, the layout and structure of an advertisement in the same newspaper can display important changes over several years. Logical layout analysis systems applied to historical documents must then account for the diachronic aspect of their layouts and adapt to the changes. Barman et al. (2020) propose a system that goes beyond usual logical labels by labelling physical block as either Serial, Weather Forecast,

Death Notice and Stock Exchange Table. To do so, their system combines visual and textual features using the word-embedding representation of each word and its coordinates on the page. Their results show that combining textual and visual features provide better results in most cases than using just one of them. Textual features are also more efficient to deal with the diachronic aspect of documents because they are more stable over time than visual features.

3 Dataset

We have processed a dataset of press and magazine documents published in the first half of the 20th century from the "Fond régional: Franche Comté" collection, available from the digital archive of Bibliothèque Nationale de France[2]. Figure 1 shows an example of the first page of a newspaper with more than 2 columns. It contains the header of the first page and several articles that contain titles and text content.

From this collection, we selected documents that had an OCR quality measure greater than 90%. This dataset was then split into a train and a test dataset. As shown in Table 2, our train set contains 15 collections of documents, which amount to a total of 48 documents, whereas the test set contains 6 collections and a total of 6 documents (Gutehrlé and Atanassova, 2021). The train and test datasets have been designed to cover as much as possible the various possible layouts that exist in the "Fond régional: Franche Comté" dataset. We have divided them into three layout types:

1c documents where the text is displayed in one column, as in books;

2c documents where the text is displayed into two columns;

3c+ documents where there are at least 3 columns of text, as in newspapers.

Table 1 shows the distribution of documents across the three layout types in our datasets.

Dataset / layout	1c	2c	3c+	Total
Train	18	5	25	48
Test	2	2	2	6

Table 1: Document layouts in the train and test datasets

[2] https://gallica.bnf.fr

The documents in the corpus cover three general topics: Catholicism, Resistance and News. The documents of the Catholic topic were published between 1900 and 1918. Most of them, such as "Bulletin paroissial de Censeau" or "Petit Écho de Sainte-Madeleine", are bulletins of small parishes. As such, they focus mainly on the local religious life, although they sometimes discuss national and international events such as WWI. The documents from the Resistance topic, such as "La Haute-Saône libre" or "La Franc-Comtoise", were published between 1939 and 1945 by Resistance fighters. As such, their main goal is to relay information about the ongoing local and international events of WWII. Finally, the documents of the News topic were published in the 1930s and focus on local and national events. Some are apolitical such as "Le Franc-Comtois de Paris", while others have a political label. For instance, "Le Semeur" and "Le Front Comtois" are left-wing newspapers whereas "Vers l'Avenir" is a right-wing Catholic newspaper.

The French language used in these documents is not very different from modern French. However, we notice some variations in the written styles between the three topics. The written style in the Catholic document is formal and literary and uses many religious metaphors. On the other hand, the written style in the News document is mostly standard, although sometimes formal. Sentences are shorter and use simpler tenses than the Catholic documents. This simplification of the writing style is even more prominent in Resistance documents. The difference in the writing style between documents can first be explained by their domain: religious text should be more literary than newspapers or Resistance periodicals. This difference can also be explained by the size of the documents. Catholic documents are the longest in the corpus, with more than 10 pages on average. As such, their text can be more elaborate. On the other hand, News and Resistance documents are respectively four and two pages long on average. Their text is factual and concise in order to convey a lot of information in the limited space they have.

All the documents are stored in the XML ALTO format, which contains descriptions of their physical layout and the text content obtained from OCR. As such, the files already provide the physical layout analysis and the reading order of the documents.

The XML ALTO format provides the text con-

Figure 1: Excerpt of the first page of the second issue of the communist newspaper *Le Semeur* published the 23rd of April 1932

Dataset	Newspapers	Issues	Text blocks	Text lines	Words	Pages
Train	15	48	4 608	51 815	338 583	368
Test	6	6	1 445	8 836	63 343	52

Table 2: Train and the test datasets

tent and physical layout of documents in the following manner. The OCR output for the whole document is available in a PrintSpace tag. Lines of text are contained in TextLine tags, which in their turn contain String tags for words and SP tags for spaces. TextLine tags are grouped into blocks in TextBlock tags. Sometimes, TextBlock tags are also grouped into ComposedBlock tags. TextBlock and TextLine tags have the following attributes:

Id the tag's identifier

Height, Width the text height and width

Vpos the vertical position of the text on the page. The higher the value, the lower the word is on the page

Hpos the horizontal position of the text on the page. The higher the value, the further on the right the text is on the page

Language the language of the text (only for TextBlock tags).

Among the attributes listed above, some TextBlock tags also have a Type attribute. This attribute is useful as it contains the logical labels of the lines in the block. It appears most often for tables or advertisements. However, TextBlock with a Type attribute are rare in our dataset. As shown in Table 3, nearly 98% of the TextBlock tags in the train and the test datasets do not have a Type attribute.

	Train		Test	
Type attribute	Count	Perc.	Count	Perc.
No attribute	4 514	97.96	1 423	98.48
illegible	79	1.71	15	1.04
titre1	15	0.33	0	0
advertisement	0	0	4	0.28
table	0	0	2	0.14
textStamped	0	0	1	0.07

Table 3: Type attribute distribution on TextBlock tags in the train and test datasets

4 Methodology

Our algorithm aims to attribute logical layout labels to both TextBlock and TextLine tags in documents. In the following subsections, we present the tagset that is used, then we explain the general processing pipeline of the algorithm. Finally, we present in detail the features that are used by the algorithm and the sets of rules. The diagram in Figure 2 shows the processing pipeline as described in this section.

We defined sets of rules for the annotation of TextBlock and TextLine tags. They are applied to documents regardless of the layout category they belong to. These rules were designed using heuristics based on observations that we made in the train

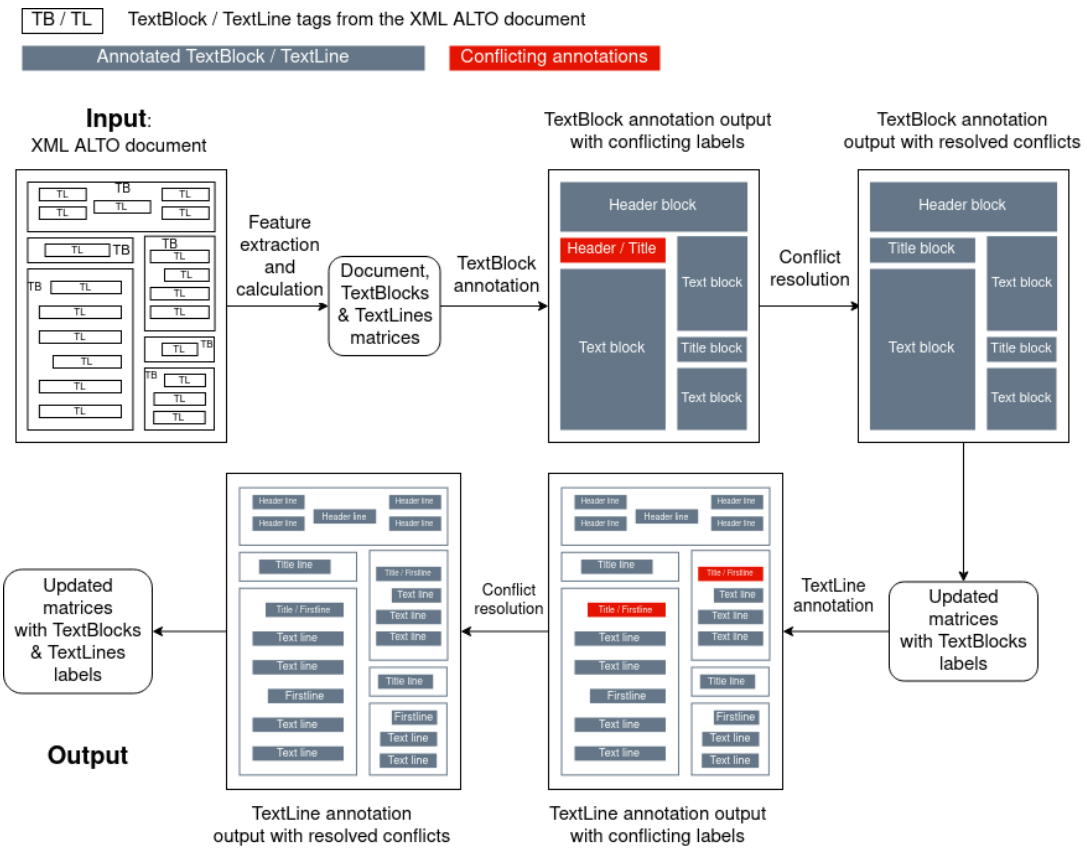

Figure 2: Diagram representation of the main stages of the algorithm

dataset. For instance, we observed that the biggest titles in the documents start with a capital letter and are surrounded by important spaces. Then, we translated these patterns into rules that we could use. This required us to extract features from the XML ALTO document, and to compute other features that are not available directly, such as the space between lines or the case of the first letter of a line.

4.1 Logical Layout Tagset

To perform the Logical Layout Analysis of the documents, we define the following annotation tagset:

- TextBlock labels: Text, Title, Header, Other;
- TextLine labels: Text, Firstline, Title, Header, Other.

The label "Firstline" must be understood as "first line of the paragraph". Thus, any TextLine tag labelled Firstline will indicate the beginning of a paragraph.

The whole dataset has been manually annotated by a single annotator, then split into a train and a test dataset. Table 4 shows the label distribution in the datasets. The train set was used to develop the rules presented in Tables 6 and 7. The test set was kept blind until the final evaluation of the system.

A small portion of the TextBlock and TextLine tags correspond to elements that are not relevant for our study, such as images, tables or advertisement. Those elements were labelled as "Other" and are ignored for the evaluation. Our system will assign the label "Other" to a TextBlock or a TextLine tag only if no other label has been assigned to it already.

4.2 General processing pipeline

The first step of our processing pipeline extracts features from the XML ALTO document at the TextLine, TextBlock and Document levels. The exact features extracted for each level are presented in Section 4.3. These features are grouped into three categories: geometric, morphological and semantic, as in Rangoni et al. (2011), Bitew (2018), Abreu et al. (2019), Tomas Hercig (2019) and Giguet and Lejeune (2019). Geometric features correspond to the physical attributes of the tags such as its height, width, or position in the document. Morphological features concern aspects of the text inside the tags,

		Train		Test	
	Label	Count	Percentage	Count	Percentage
TextLine	Text	36 272	70.138	6 648	75.881
	Firstline	9 785	18.921	1 563	17.840
	Title	1 820	3.519	234	2.670
	Header	740	1.430	115	1.312
	Other	3 098	5.989	201	2.293
TextBlock	Text	2 064	45.724	1 102	80.203
	Title	429	9.503	90	6.550
	Header	333	7.377	53	3.857
	Other	1 686	37.35	128	9.314

Table 4: TextBlock and TextLine tags label distribution over the train and test datasets

for instance finding if a line starts with a capital letter or a digit. Finally, semantic features concern the content of the tag, like the presence of a specific keyword. We store these features into two matrices for TextLine and TextBlock features and in a dictionary for Document features. Each row in the matrices represents either a TextLine or a TextBlock tag and each column is a corresponding feature.

The second step attributes logical labels to TextBlock tags. Labelling TextBlock before TextLine is important because the presence of a Type attribute in TextBlocks can help label the lines inside these blocks. The goal of this step is to add a Type attribute to every TextBlock. To do so, we process the TextBlock feature matrix from the previous step by applying sets of annotation rules, one for each possible logical label. A TextBlock is only processed if it doesn't already have a Type attribute. Because the sets of rules are applied independently from each other, a same TextBlock can obtain multiple labels. Another set of rules is then applied to solve such conflicts and keep only one possible logical label for each TextBlock, which is then set as the value of the Block's Type attribute in the feature matrix. The complete sets of rules to annotate TextBlock tags and solve conflicts are presented in Section 4.4.

The third step attributes logical labels to TextLine tags. Every TextLine is by default labelled as Text. The system then applies rules to identify the other labels. First, any TextLine in a Title or a Header block inherits the same label. Then, any TextLine contained in a TextBlock is processed by a set of rules in order to identify Firstlines and possible missing Titles. Similarly to the previous step, rules are applied independently from each other, resulting sometimes in conflicting predictions. The TextLine feature matrix is processed a second time to solve conflicting predictions and keep only one possible label for each TextLine tags. This step also controls that any line that follows a Title is labelled as Firstline and that the first line of the document is labelled as Title if it not already labelled as Header. The complete sets of rules to annotate TextLine tags and solve conflicts are presented in Section 4.5.

The algorithm finally outputs the three feature matrices, where the TextBlock and TextLine matrices have been updated with the annotations of both steps 2 and 3.

4.3 TextBlock, TextLine and Document features

Our algorithm uses sets of features that are extracted and calculated from the XML ALTO document at three different levels: TextLine, TextBlock and Document level. Table 5 presents all the features with their descriptions and levels. The information on these features for all document elements, in the form of matrices, is the input of the annotation rules that are described in the following subsections.

The header words set, which is used for the calculation of the *simHeaderSet* feature, is made up of the following words or phrases: *Rubrique Locale, Gérant, Publicité, Abonnement, Envoyez les fonds, Conservez chaque numéro, Rédacteur, Directeur, Numéro, Chèque postal, Dépôt, Achat-Vente-Echange, Annonce, Imprimerie, En vente partout, Paraissant*. This list is necessary for the annotation of the TextBlocks that represent the headers of the newspaper pages. It has been created by observing the different types of headers that exist in the datasets.

4.4 TextBlock annotation rules

The annotation rules that we have defined use sets of conditions that must be verified on the features of the TextBlock elements. All the rules are applied

Feature	Description	TextLine	TextBlock	Document
page	page number of the page containing the element	X	X	
blockType	type of the block	X	X	
wordCount	number of words	X	X	
precedingSpace, followingSpace	spaces between the element and those before and after it	X	X	
capitalPro, digitProp	proportion of capital letters and digits	X	X	
height, width	height and width values of the line	X		
hpos, vpos	coordinates of the line on the page, i.e. its horizontal and vertical position	X		
diffHpos	the difference between *hpos* and the median *hpos* value in the block	X		
stwCapital, stwDigit	True if the line starts either by a capital letter or a number, False otherwise	X		
headerMark1	True if the element contains the word "Page" or a dash sign. False otherwise.	X	X	
headerMark2	True if the element contains a date, a currency, an address. False otherwise.	X	X	
simTitle	similarity of the line with the title of the document, calculated by the Levenshtein distance	X		
simHeaderSet	highest similarity of the line with the words contained in the header words set, calculated by the Levenshtein distance	X		
firsthpos, firstvpos	coordinates of the first line of the block		X	
lasthpos, lastvpos	coordinates of the last line of the block		X	
linecount	number of lines		X	
medHeight, medWidth	median line height and line width		X	X
medHpos, medVpos	median *hpos* and *vpos* values in the block		X	
medWordCount, medLineSpace	median number of words by line and the median space between lines in the block		X	X
wordRatio	number of words by line		X	
medBlockHeight, medBlockWidth	median line height and block height and width			X
medBlockSpace	median space value between blocks			X
thirdQuartileLineSpace	third quartile of line space values in the document			X
medWordRatio, medLineCount	median number of words by line and median number of line by block in the document			X

Table 5: List of features used by the algorithm

to all TextBlock tags in the documents. Identifying Text and Title blocks relies on geometric and morphological features, whereas identifying Header blocks relies on semantic features.

Text blocks contain relatively more lines and more words than other blocks in the document. Title blocks are TextBlock tags that contain few lines, usually not more than 3. The role of a title is to introduce the topic of a text section, thus a Title block should be surrounded by Text blocks. The space around that block should also be important, in order to stand out with the surrounding blocks. A Text block should have a smaller height than a Title block. As such, if there is a confusion between Text and Title block, we use the height of the block to distinguish between the two.

Headers contain very specific information about the document, such as its title, its price, a date or the publisher's name. This information is displayed with keywords and sentences that are recurrent across multiple pages and documents. As Header blocks are only located at the top of a page, we only look for this information in the first four lines of a page. Small blocks at the top of a page are most likely Headers. Considering the first page, we look for the header in the first 30 lines, because the first page's header contains more information.

Table 6 presents all annotation rules for TextBlock tags and their corresponding annotation labels, where B is a TextBlock in a document D. The last two rules, 6 and 7, solve conflicting annotations.

4.5 TextLine annotation rules

Naturally, TextLine tags that are contained in a Title or Header block inherit this annotation. TextLine tags that appear between two Header lines are also annotated as Header. To find Firstline and missing Title lines, we apply sets of rules that rely on geometric and morphological features.

TextLines inside Text blocks are processed in order to identify Firstlines and possible missing Titles. The first line of a paragraph always starts with a capital letter, and most of the FirstLine are

Rule	Condition	Label
1	*(B.linecount > D.medLineCount)* or *(B.wordCount > D.medWordCount/3)*	Text
2	Previous and next TextBlocks are Text and *(B.linecount < D.medLineCount)* and *(B.medHeight < D.medBlockHeight)*	Text
3	Previous and next TextBlocks are Text and B is not Text and *(B.linecount < 4)* and *(B.precedingSpace > D.medBlockSpace) or (B.followingSpace > D.medBlockSpace)*	Title
4	*B.page = 1* and for any of the first 30 lines of B: *simHeaderSet > 0.9* or *simTitle > 0.9* or *headerMark1* or *headerMark2* or *ctnTotal*	Header
5	*B.page > 1* and for any of the first 4 lines of B: *simHeaderSet > 0.9* or *simTitle > 0.9* or *headerMark1*	Header
6	Conflicting annotation: Header and (Text or Title):	
	(B.linecount < 15) and *(B.wordCount < 50)*	Header
	Otherwise	Text / Title
7	Conflicting annotation: Text and Title:	
	B.medHeight > D.medBlockHeight / 2	Title
	Otherwise	Text

Table 6: TextBlock annotation rules and conflict resolution rules

indented. For this reason, we select TextLines that have a Hpos value greater than the other TextLines in the block. The Firstlines that are not indented can be identified if the line that precedes them is shorter, indicating the end of the previous paragraph. Finally, the first line of a page or immediately after a Title is labelled Firstline, if it starts with a capital letter.

Like Title blocks, Title lines are surrounded by relatively more space in order to stand out from other text sections. The smaller the title is, the less important the space around it is. Small titles usually contain more capital letters and are center-aligned. Thus, all these criteria enter consideration for the identification of Titles.

Table 7 presents the TextLine annotation rules and their corresponding annotation labels, where L is a TextLine in a document D and B is the TextBlock that contains L. The last two rules, 11 and 12, solve conflicting annotations.

5 Evaluation

To evaluate our method and the proposed annotation rules we have run the algorithm through the test dataset. Table 8 shows the Precision, Recall and F1 scores for the TextBlock and TextLine classification steps.

TextBlock annotation is an intermediary step in the algorithm. TextBlock annotation rules perform best on documents from the 2c layout category. Title classification for TextBlocks performs with F1 score of 0.61 on average and 0.94 on documents from the 2c category. Header classification for TextBlocks provides a good precision score (0.726) but with a low recall (0.298).

Similarly to TextBlock annotation rules, TextLine annotation rules perform best on documents from the 2c category. Title identification performs worse on 1c documents, and obtains overall F1 score of 0.639 for all layouts. Firstline identification performs fairly well with an F1 score above 0.9. Header identification obtains a good precision score (0.803) but with a recall of 0.348. This means that header identification rules are insufficient and need to be completed to capture the various types of headers.

A first type of error comes from errors in the Block classification step. As any line in a Title or Header block inherits that annotation, the precision of TextBlock annotation is an important factor for the overall performance of the algorithm.

A second type of error is the confusion between Titles and First lines. Most Titles mislabelled as Firstline are short subsection titles. As such, they are similar to other text lines in terms of typography, and are hard to detect with the features we use. This confusion happens mainly in documents from the 2c and 3c+ categories. Other mislabelled Titles are one-line paragraphs such as greetings or signatures, or the beginning of a text section. Such lines have properties similar to Titles, being surrounded by important spaces and being either center or right-aligned. Extracting features about the font style of the line (bold, italics) and its alignment (left, center, right-aligned) could help solve this confusion.

6 Conclusion and Discussion

In this article, we have presented a rule-based system for the Logical Layout Analysis of XML ALTO documents. Our system starts by extracting features from the document, then uses these features to add logical labels to TextBlock and TextLine tags. We have described the construction and the evaluation of the proposed annotation

Rule	Condition	Label
1	L.precedingSpace = 0 and L.followingSpace > D.medLineSpace and L.simTitle < 60 and L.simHeaderSet < 60 and L.stwCapital	Title
2	L.wordCount < B.medWordCount and L.precedingSpace > D.thirdQuartileLineSpace and L.followingSpace > D.thirdQuartileLineSpace	Title
3	L.capitalProp > 10 and L.wordCount < B.medWordCount and L.height < B.medHeight and (L.precedingSpace > D.thirdQuartileLineSpace or L.followingSpace > D.thirdQuartileLineSpace)	Title
4	L.diffHpos > 104 and L.capitalProp > 0 and L.precedingSpace > D.medLineSpace and L.followingSpace > D.medLineSpace	Title
5	L.hpos > B.medHpos and L.diffHpos < 105 and (L.stwCapital or L.stwDigit)	Firstline
6	L.width < B.medWidth and L.wordCount < B.medWordCount and L.Hpos < B.medHpos	Lastline
7	Previous TextLine is LastLine and L.stwCapital and L.followingSpace < B.medLineSpace	Firstline
8	Previous TextLine is not Lastline and L.stwCapital and L.precedingSpace > B.medLineSpace and L.followingSpace < B.medLineSpace	Firstline
9	Previous TextLine is not Lastline and L.stwCapital and L.hpos > B.medHpos	Firstline
10	None of the rules 1-9 above is True	Text
11	Conflicting annotation: Header and other label: Previous TextLine is Header and next TextLine is Header	 Header
12	Conflicting annotation: Title and FirstLine: L.followingSpace < B.medLineSpace and L.capitalProp < 15 Otherwise	 Title Firstline

Table 7: TextLine annotation rules

		Text			Title			Firstline			Header		
	Cat	P	R	F1	P	R	F1	P	R	F1	P	R	F1
TextBlock	1c	0.947	0.938	0.942	0.312	0.357	0.333				0.679	0.373	0.476
	2c	0.973	0.989	0.981	0.899	1.000	0.947				1.000	0.271	0.411
	3c+	0.958	0.973	0.965	0.589	0.560	0.551				0.500	0.250	0.333
	Mean	0.959	0.966	0.962	0.600	0.639	0.610				0.726	0.298	0.406
TextLine	1c	0.979	0.986	0.983	0.354	0.720	0.473	0.943	0.854	0.895	0.909	0.598	0.721
	2c	0.961	0.995	0.978	0.746	0.765	0.747	0.955	0.859	0.902	1.000	0.118	0.197
	3c+	0.975	0.992	0.983	0.703	0.702	0.702	0.952	0.877	0.913	0.500	0.400	0.444
	Mean	0.969	0.991	0.979	0.595	0.733	0.639	0.949	0.861	0.902	0.803	0.348	0.435

Table 8: Precision, Recall and F1-score for TextBlock and TextLine annotation

rules. This methodology provides very good results for some categories like Text, Firstlines in most cases, but struggles with other labels such as Headers or Titles. Most errors in our system can be corrected by either adding new rules or by refining the already existing ones. The system could also benefit from adding new features such as font style and line alignment.

While recent methods in NLP use extensively machine learning and deep learning architectures, such approaches require large annotated datasets. To the best of our knowledge, no such datasets exist for the logical layout analysis of historical newspapers in French. For this reason, the algorithm that we propose in this paper is manually designed and rule-based. Its objective is, above all, to be able to produce annotated datasets that are large enough to envisage machine learning or deep learning approaches. The comparison between the performance of these rules and the results of recent deep learning architectures will be the object of our future work.

We devised the rules to process documents regardless of their era. As stated earlier, the layout in historical documents evolves rapidly, especially in newspapers. In order to create sets of rules dedicated to the different publication periods, we plan in future works to apply rule learning algorithms to generalise the creation of rules.

Acknowledgments

This research is supported by the Région Bourgogne Franche-Comté, France, as part of the EMONTAL project (2020–2024).

References

Carla Abreu, Henrique Lopes Cardoso, and Eugénio Oliveira. 2019. Findse@fintoc-2019 shared task.

Hanna Abi Akl, Anubhav Gupta, and Dominique Mariko. 2019. FinTOC-2019 shared task: Finding title in text blocks. In *Proceedings of the Second Financial Narrative Processing Workshop (FNP 2019)*, pages 58–62, Turku, Finland. Linköping University Electronic Press.

Raphaël Barman, Maud Ehrmann, S. Clematide,

S. Oliveira, and F. Kaplan. 2020. Combining visual and textual features for semantic segmentation of historical newspapers. *ArXiv*, abs/2002.06144.

Semere Kiros Bitew. 2018. Logical structure extraction of electronic documents using contextual information.

Marius Bulacu, Rutger van Koert, Lambert Schomaker, and Tijn van der Zant. 2007. Layout analysis of handwritten historical documents for searching the archive of the cabinet of the dutch queen. In *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, volume 1, pages 357–361.

Kai Chen and Mathias Seuret. 2017. Convolutional neural networks for page segmentation of historical document images.

Christian Clausner, Christos Papadopoulos, Stefan Pletschacher, and Apostolos Antonacopoulos. 2015. The enp image and ground truth dataset of historical newspapers. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, pages 931–935.

Emmanuel Giguet and Gaël Lejeune. 2019. Daniel@FinTOC-2019 shared task : TOC extraction and title detection. In *Proceedings of the Second Financial Narrative Processing Workshop (FNP 2019)*, pages 63–68, Turku, Finland. Linköping University Electronic Press.

Nicolas Gutehrlé and Iana Atanassova. 2021. Dataset for Logical-layout analysis on French historical newspapers.

Marti A. Hearst. 1997. Texttiling: Segmenting text into multi-paragraph subtopic passages. *Comput. Linguist.*, 23(1):33–64.

Reza Farrahi Moghaddam Hossein Ziaie Nafchi, Seyed Morteza Ayatollahi and Mohamed Cheriet. 2012. Persian heritage image binarization dataset (phibd 2012).

David Hébert, Thomas Palfray, Stéphane Nicolas, Pierrick Tranouez, and Thierry Paquet. 2014. Automatic article extraction in old newspapers digitized collections. *ACM International Conference Proceeding Series*.

K. Kise, M. Iwata, and Keinosuke Matsumoto. 1999. On the application of voronoi diagrams to page segmentation.

S. Klampfl and Roman Kern. 2013. An unsupervised machine learning approach to body text and table of contents extraction from digital scientific articles. In *TPDL*.

G. Nagy, S. Seth, and M. Viswanathan. 1992. A prototype document image analysis system for technical journals. *Computer*, 25:10–22.

Anoop Namboodiri and Anil Jain. 2007. *Document Structure and Layout Analysis*, pages 29–48.

D. Niyogi and S.N. Srihari. 1995. Knowledge-based derivation of document logical structure. In *Proceedings of 3rd International Conference on Document Analysis and Recognition*, volume 1, pages 472–475 vol.1.

L. O'Gorman. 1993. The document spectrum for page layout analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 15:1162–1173.

Cartic Ramakrishnan, Abhishek Patnia, Eduard Hovy, and Gully Burns. 2012. Layout-aware text extraction from full-text pdf of scientific articles. *Source code for biology and medicine*, 7:7.

Y. Rangoni, A. Belaïd, and Szilárd Vajda. 2011. Labelling logical structures of document images using a dynamic perceptive neural network. *International Journal on Document Analysis and Recognition (IJDAR)*, 15:45–55.

Martin Riedl, Daniela Betz, and Sebastian Padó. 2019. Clustering-based article identification in historical newspapers. In *Proceedings of the 3rd Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 12–17, Minneapolis, USA. Association for Computational Linguistics.

Zejiang Shen, Kaixuan Zhang, and Melissa Dell. 2020. A large dataset of historical japanese documents with complex layouts. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2336–2343.

Fotini Simistira, Mathias Seuret, Nicole Eichenberger, A. Garz, M. Liwicki, and R. Ingold. 2016. Divahisdb: A precisely annotated large dataset of challenging medieval manuscripts. *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 471–476.

H. Tibbo. 2007. Primarily history in america: How u.s. historians search for primary materials at the dawn of the digital age. *American Archivist*, 66:9–50.

Pavel Král Tomas Hercig. 2019. UWB@FinTOC-2019 shared task: Financial document title detection. In *Proceedings of the Second Financial Narrative Processing Workshop (FNP 2019)*, pages 74–78, Turku, Finland. Linköping University Electronic Press.

Xu Zhong, J. Tang, and Antonio Jimeno-Yepes. 2019. Publaynet: Largest dataset ever for document layout analysis. *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1015–1022.

Annus Zulfiqar, Adnan Ul-Hasan, and Faisal Shafait. 2019. Logical layout analysis using deep learning. In *2019 Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–5.

"Don't worry, it's just noise": quantifying the impact of files treated as single textual units when they are really collections

Thibault Clérice
Centre Jean Mabillon, École Nationale des Chartes, PSL University / 65 rue Richelieu, 75002 Paris, France
HiSoMa, Université Lyon 3
thibault.clerice@chartes.psl.eu

Abstract

Literature works may present many autonomous or semi-autonomous units, such as poems for the first or chapter for the second. We make the hypothesis that such cuts in the text's flow, if not taken care of in the way we process text, have an impact on the application of the distributional hypothesis. We test this hypothesis with a large 20M tokens corpus of Latin works, by using text files as a single unit or multiple "autonomous" units for the analysis of selected words. For groups of rare words and words specific to heavily segmented works, the results show that their semantic space is mostly different between both versions of the corpus. For the 1000 most frequent words of the corpus, variations are important as soon as the window for defining neighborhood is larger or equal to 10 words.

1 Introduction

"You shall know a word by the company it keeps."(Firth, 1957). Over the last decades, Firth's sentence has seen its frequency grow as tools for analyzing both short and large corpora have found their way into the personal computer of students and researchers in other fields than linguistics, corpus linguistics and natural language processing in general (Heiden, 2010; Sinclair and Rockwell, 2016). Research using this postulate, efficiently summarizing the semantic distributional hypothesis[1], have been used in both the Latin and Ancient Greek domains on classical (Roda et al., 2019), late antiquity (Munson, 2017), medieval (Guerreau, 1989; Perreaux, 2012) and post-medieval literature (Bloem et al., 2020) with corpora spanning from few dozen thousand tokens to a few millions. As for many experiences, the set-up of this kind of information extraction[2] might have an important impact on the outcome of the analysis, as it will influence the "companionship" of the analyzed words: normalizing the text with lemmatization for example will reduce complexity and augment the signal for morphologically rich languages but reduce details found in some forms (*e.g.*, imperative in verbs); for the same reason, manipulating the size of the window defining the neighborhood of words allows for capturing more or less information. If these pre-processing steps are often known and made explicit, one is often unclear or dismissed: the way the source corpus is encoded and read is often omitted by authors[3]. But what happens when someone uses a digitized corpus made up of composite works?

If we were to take the example of the Perseus's "Canonical Latin Literature" repository (Crane, 2021b), some files are actually composite works, such Martial's *Epigrammata*, a collection of more than a thousand poems of varying but short lengths, while some others are "more" monolithic such as an *oratio* of Cicero. While there are (rare) studies of noise in digital humanities, and the few that exist focus on OCR quality and its impact (Eder, 2013a,b), none seems to have addressed the potential impact of treating texts as continuous strings of tokens, as they are found in digital format (plain text or the like) instead of treating them as collections of independent textual units within the same file. We note, however, the study of (Schöch, 2017) which explicitly studies two forms of segmentation, plays as a whole and arbitrary segmented plays, and their effect in a bag of words approach.

[1] Based on this hypothesis, we expect words sharing the same meaning or being semantically close to each other to be found with the same neighbor words. *e.g.*, "This morning, I drank an _ juice" is easy to fill with multiple words (*e.g.*, orange, apple) which share the same semantic trait: they are fruits.

[2] And others using bag of words such as topic modeling.

[3] *e.g.*, neither (Köntges, 2020) – in the context of a Bag-of-words approach – nor (Stringham and Izbicki, 2020) – in the context of word embeddings – address this question.

Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH), pages 95–105
December 19, 2021.

2 Corpus, concepts and methodology

2.1 Concepts for segmentation

In Classical Latin, as in most modern books, published works are usually split in various smaller textual units, which might be chapter, recipes, poems, etc. For work in prose such as novels or history books, chapters and paragraphs are usually the unit one could refer to. This segmentation is often an editorial or authorial way to indicate from light to strong topical shifts or narrative ellipsis. In poetry, most poems are published in form of collections, and, at least for Latin literature, they are not expected to be sequential: there are very few if none that connect throughout Martial's *Epigrammata* as direct sequence, and when there are connections, they probably are more echoes than the result of a progression. There are other genres and textual forms that we were able to keep over millennia, such as Apicius' *Recipes* to medicine notebooks such as Caelius Aurelianus' *Gynaeciorum Sorani* or grammatical commentaries from scholiasts such as Porphyrio: again, these are merely a single cohesive narrative sequence but rather a collection of short units, connected through a global theme.

And unlike modern literature, where we would expect chapters and paragraphs to be authorial marks on the text, the status of these marks can differ from one genre to another for ancient literature. These texts have been transmitted, reinterpreted and – as such – modified as soon as few centuries after they were first published[4]. For some of these segmentations, we know for a fact they were there originally: this is the case for the segmentation category we call "book" today which were often rolls or *volumen* published by authors at the time (Canfora, 2016, p. 13). For poetry, most of the segmentation in poems is certainly drawn from the original work with some doubts for the order of poems[5]: for rarely copied works such as the *Priapea*, some doubts can be easily instigated in how some poems can be segmented, but it remains a rare case. On the other hand, we know for a fact that some works were cut or reorganized by latter hands, such as *scholia* and commentaries in general: current hypothesis have them originating as notes to a text connected through lemma (*hypomnemata*) or *glosae*, and ended up as continuous texts in which the text was inserted (Bureau, 2012). In most other situations, the current segmentation of the text is either the effect of medieval scholars, such as for the verse numbering of the Bible, 16–17th century editors or modern ones: such is the case for the *Pro Murena*, as Fotheringham demonstrates it (Fotheringham, 2007). Not only the later text exists with two competing segmentation, but the paragraphs, when they are not numbered and identified, are sometimes not the same from one editor to the other. Of course, there are even more complex textual traditions which sometimes challenge text order, such as the one of Petron's *Satyricon* or Plaute's plays, and propose completely different forms of works, such as the *Epistola Alexandri ad Aristotelem*, an anonymous work which exists in two different *recensio*.

Whoever segmented the texts, authors or editors, they carry information about how the full work should be read by a human being. In this context, we propose to categorize the units formed by these segmentation in two types: on one hand, the ones that are clearly non-sequential – such as poems – as *autonomous textual units* (ATU), on the other, the more loosely connected elements – such as chapters – as *Semi-Autonomous Textual Units* (SATU). In this context, textual autonomy is achieved when a word from a textual unit and the word from following or preceding units cannot be classed as co-occurring, such as poems, because they are narratively, thematically or discursively unrelated. For SATU, the semi-autonomous character can be discussed, but chapters or books certainly would display a certain level of discursive autonomy with each other, while enabling discursive progression. In Latin corpora such as the ones following CapiTainS encoding guidelines for TEI (Clérice, 2017), each text has been thoroughly annotated with a citation scheme, such as Book → Poem → Line, by their corpus editorial team.

As these texts could be used within the framework of the distributional hypothesis, we propose a first metric to evaluate the potential risk of noise that would be introduced using fixed windows for context retrieval: the *theoretical window contamination rate*. For a given text t, it can be quantified as a function of the number of (S)ATU of the text

[4] And for some of the work we know under a single author's name and a single work title, we know for a fact there was either multiple authors (*e.g.*, Caesar's *De bello gallico*), multiple original works collected by later "editors" (*e.g.*, the Bible) or both such as Sulpicia whose elegies are found in the *Corpus Tibullianum*.

[5] See the difference in the edition of Leon Herrmann (Catulle and Herrmann, 1957) compared to the others such as Lafaye's one (Catulle and Lafaye, 1932).

(Poem 39)
Iliaco similem puerum, Faustine, ministro
Lusca Lycoris amat. Quam bene lusca videt!
(Poem 40)
Inserta phialae Mentoris manu **ducta**
Lacerta vivit et timetur argentum.
(Poem 41)
Mutua quod nobis ter quinquagena dedisti
Ex opibus tantis, quas gravis arca premit,
[...]

(Poem 39)
Iliaco similem puerum, Faustine, ministro
Lusca Lycoris amat. Quam bene lusca videt!
(Poem 40)
Inserta phialae Mentoris manu **ducta**
Lacerta vivit et timetur argentum.
(Poem 41)
Mutua quod nobis ter quinquagena dedisti
Ex opibus tantis, quas gravis arca premit,
[...]

Figure 1: Book 3 Poem 39–41 from Martial's *Epigrammata*. The co-occurring words of **ducta** for $W = 10$ are underlined, on the left in a segmented corpus, on right in a raw corpus.

$|U_t|$, the size of the window used for semantic information retrieval W and the number of tokens in the text $|t|$ such as

$$Rate(t) = \begin{cases} \frac{2W(|U_t|-1)}{|t|} & \text{if } |t| > 2W \\ 0 & \text{otherwise} \end{cases}$$

where each token in a (S)ATU until W has up to $2W$ co-occurring tokens drawn from neighbor (S)ATU except for the very first and last units ($|U| - 2 \times \frac{1}{2}$) of the text, which has either no following or no preceding unit (hence $\frac{1}{2}$). This rate represents the relative quantity of tokens whose window has at least one token not supposed to be counted as co-occurring. In this context, with a default window of 5-words of tools such as Gensim (Řehůřek et al., 2011), we have high rates for texts such as 21.22% for Martial's *Epigrammata*, a collection of 1,527 poems over 14 books and 71,911 tokens, and 0% for work in prose such as Sallust's *Iugurthia* (continuous which means only 1 (S)ATU while having 25,411 tokens). This would imply that up to one fourth of the tokens of Martial's work could end up polluted by non-co-occurring words, albeit at various scales (the first and last words of each unit being more polluted than the $W + 1$ one which end up with only 1 noise token). However, the theoretical window contamination rate assumes an equally distributed number of words in (S)ATU and is an efficient tool to consider the issue, the real contamination rate being dependent of the size of previous passages, following passages and size of each (S)ATU. In the case of very small poems such as Martial's *Epigrammata* 3.40 (10 tokens), not a single token window contains a clean set of co-occurring words starting with $W = 5$, with a real contamination rate reaching 1.0, and from $W \geq 10$, each token draw co-occurrences from both neighbor units at the same time (*cf.* Figure 1).

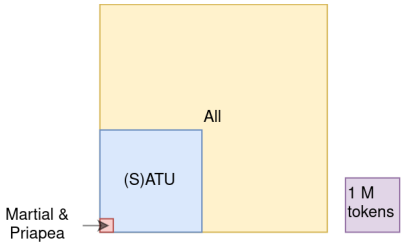

Figure 2: Scaled representation of the three sub-corpora and their relation to each other: each corpus contains the smaller one(s).

2.2 Corpora and Their Pre-Processing

In order to evaluate the effect of dismissing (S)ATU's importance, we propose a study based on the *Corpus Latin antiquité et antiquité tardive lemmatisé* (Clérice, 2020a) which aggregates original works from Perseus (Crane, 2021b), Open Greek and Latin (Crane, 2021a), Lasciva Roma (Clérice, 2020b, 2021a) and DigilibLT (Lana, 2021; Clérice et al., 2021). The lemmatization of this corpus and its pre-processing was applied with Pie-Extended's LASLA model (Manjavacas et al., 2019; Clérice, 2020c, 2021b) which has a 97.34% accuracy. Overall, the aggregated corpus spans from 254 BCE to 799 CE with 21,222,911 tokens - including punctuation, after tokenization and lemmatization - over 853 works[6], composite or not (some being collections of works of multiple authors, sometimes multiple unidentified ones, such as the *Anthologia Latina*). Each source corpus was encoded – at least partially – with CapiTainS guidelines (Clérice, 2017), allowing for machine-actionable segmentation in TEI documents and as such allowing us to retrieve or segment whole works with their editorial segmentation, such as poems, lines, books, etc.

We then divide the main corpus in three sub-corpora (*cf.* Figure 2):

1. The first corpus contains only Martial's *Epigrammata* and the anonymous *Priapea* (*Martial & Priapea* hereafter): they form a very small corpus ($|t| = 61,082$), with shared topics and vocabulary (they are full of sexual and obscene words) and are heavily composite. In fact, the first is separated in three levels (book,poem,line), the latter in only two levels (poem,line). Both text provide ATU levels (poems).

2. The second (*ATU Corpus* hereafter) consists

[6] 17,804,769, excluding punctuation.

of all works in which there is a unit level qualified as "poem", "comment" or "scholia", "letter", "speech" and "entry" (found dictionaries-like works or collection of recipes): this amounts to 125 works and 3,549,249 tokens. They do not share any specific topical unity but are massively consisting of poetry. It also is a superset of *Martial & Priapea*.

3. The last consists of the full corpus with its 17,639,626 tokens (*All*) after removing punctuation and other foreign tokens.

Each corpus displays a different value of the combined property "Number of (S)ATU-Corpus size" (*cf.* Table 1): the first is a very small corpus with a high number of autonomous textual units, the second has a high number of ATU but reaches a bigger number of tokens (nearing 1.5 million) that might be better at dealing with noise while the last is a mixture massively built of long passages in small amount per work with the exception of its contained sub-corpus *ATU Corpus* and some other texts that could not be easily fit into the latter automatically.

For each corpus, in order to quantify the effect of segmentation or its absence, each text was assigned a level at which they should be split allowing (S)ATU to be treated as non-sequential units: for texts which do not show any (S)ATU, the full text was kept as a single unit. We produce two versions of our corpora: the first one where (S)ATU are used to prevent window from overreaching, which we call segmented corpus $S(T)$, and the second where each file is treated as a single continuous unit of information, called unsegmented corpus $U(T)$.

2.3 Semantic analysis and experimental set-up

As the main objective is to analyze the impact of text segmentation on semantic analysis, we set up the experiment based on four parameters which we then combine to produce analysis using both versions of T and compare their results.

The first parameter set is composed of four groups of words we want to analyze in the corpus. These sets of words, which we call pivots, provide different distribution depending on the corpora and are composed of words appearing at least 10 times:

1. The first one, *Puer et al.*, contains words related to people and is not specific to any of the corpora. These are *dominus, mater, pater, puella, puer, uir, uxor*. They span from 2,036 occurrences (*puella*) to 48,519 (*dominus*) in *All*.

2. The second, *Carmen et al.*, might be more specific to grammarians and poetry, which are overrepresented in *ATU Corpus*. It contains *scribo, poeta, libellus, lego2, carmen1, liber1* (*cf.* Table 2)[7].

3. The third, *Puer, Carmen et al.*, is a combination of the first two and offers as such two clearly separated semantic subgroups which should be easy to cluster when time comes.

4. The last, *Futuo, Carmen, Puer et al.*, is a combination of the first two as well as crude words and words which are connected to sexuality: for some, they are heavily specific to *Martial and Priapea*, have a very low frequency compared to the first group, but are also somewhat specific to *ATU Corpus*. They are *cunnus, fello, futuo, irrumo, lasciuus, mentula, paedico2* (*cf.* Table 2 for each word frequency depending on the corpus).

5. In order to evaluate noise before analysis, we also consider a fifth word-set made of the 1000 most frequent words of the *All* dataset.

A second parameter is the size of the window, written W. To analyze words, we will only retrieve words occurring in this window. We make this window vary between four values ($[5, 10, 15, 20]$).

A third parameter is the floor-threshold frequency, noted F thereafter. Lemma co-occurring with our pivots will only be considered if they occur at least F time in the corpus of windows: if `lemma1` appears $F - 5$ times with `pivot1` and 5 times with `pivot2`, it is kept as a feature. F varies within $[1, 5, 10, 20]$ which should provide situations less prone to noise: unique co-occurrences will be ignored when $F > 1$ for example, and less important lemma will follow as we raise the value.

The following workflow is then applied to the first four word sets[8] using all combinations of the W and F for a total of 16 different results per version of the corpus:

[7] When lemmas are ending with numbers such as *lego2*, it represents a disambiguation index: in Latin, the first person of indicative present is often used to represent lemma, but two verbs share the *lego* form: one is conjugated *legis* at the second person (meaning: read, *lego2*) while the other becomes *legas* (meaning: name, *lego1*)

[8] *Top1000* is not used in the whole experiment, see below.

	(S)ATU		Tokens		Texts		Distribution of Tokens / ATU						
	Count	%	Count	%	Count	%	Mean	Std	Min	25%	50%	75%	max
Martial & Priapea	1607	2.8	61,056	0.3	2	0.2	38	32	7	13	27	54	280
ATU Corpus	39,591	68.5	3,549,249	20.1	125	14.8	90	603	1	9	17	36	47,783
All	57,761	100.0	17,639,626	100.0	845	100.0	305	2,159	1	11	24	77	248,564

Table 1: Properties of the different corpora. Punctuation and foreign tokens are ignored in the token count.

1. We retrieve and store the co-occurring count in a matrix where co-occurring words constitute features (columns) and pivot classes (lines), with their number of retrieval as values. We use the output of this retrieval in section 3.1 to analyze raw variation between $U(T)$ and $S(T)$.

2. Following the work of Evert (Evert, 2005), A. Guerreau (Morsel, 2015) and N. Perreaux (Perreaux and rey, 2013), we apply a normalization algorithm called *Dice* coefficient.

3. For each pivot, we keep their 5 most correlated features (retrieved lemma in the window). If the score of the fifth word is shared by multiple words, we keep all of them. They constitute a second set of words we call *major co-occurrences* (M).

4. We retrieve and augment the original matrix in step 1 with the same retrieval and store strategy for each word in M. We use M in section 3.2 to study the impact, post-normalization, of this first step of analysis.

5. We normalize again the output with Dice coefficient: the final output here constitutes our analysis input.

This approach using bag-of-words and normalization is preferred in the context of our experiment to deep learning approaches such as Word2Vec. Given their instability in "small" corpora ($\tilde{2}$0 million)(Antoniak and Mimno, 2018) and the risk of not controlling perfectly the randomizing seed at the library (*e.g.*, *Gensim*) or Python level, we preferred non-random approaches, as any variation due to randomization, including the order in which texts are seen, might affect the results and hide the hypothetical window noise in its own randomness.

Once we have a matrix with co-occurrences count, we can then perform an analysis: while performing Dice is a first step of post-processing, we want to see how the data would react to traditional means of analysis. We preferred in this context to use Ward agglomerative clustering using Euclidean distance on pivots and major co-occurrences. In order to have the ability to compare the output of the clustering on $U(T)$ and $S(T)$, we harmonize major co-occurrences by stripping the ones which are not shared between analysis running with the same parameters over both versions of T. It produces a set of $|C|$ common classes which can finally be clustered according to a fifth parameter k, where k is the number of clusters we want to obtain. We make k vary so that $\frac{|C|}{k} < 2$ and $5 \geq k < 15$. As studying the variation is the objective of this paper, the number of clusters does not need to be fine-tuned, as we are only interested in the equality or inequality between the analysis of $U(T)$ and $S(T)$, thus varying k within a dynamic range. The output of this final step is finally studied in section 3.3.

3 Evaluation of impact

Once all combinations have been run, we want to evaluate three different kinds of differences or effects: a first raw effect on lemma co-occurrences and how the raw matrices would differ without any normalization step, the second effect on the selection of secondary classes (major co-occurrences) and finally the effect on more advanced analysis, here using clustering, through the evolution of features.

3.1 On the Co-occurrence Matrix

In order to quantify the impact on neighborhood retrieval, we propose to first analyze the impact on the most frequent words of the corpus that are either adverbs, adjectives, pronouns, nouns or verbs. Then, for each corpus, we run the step 1 described above for each combination of W and F. We compute for each lemma the Manhattan distance between its vector in the result matrices of both $S(T)$ and $U(T)$ where each absent feature is replaced by a column filled with 0.

While some lemmas do not show any variation between versions of T, a vast majority of them displays non-null distances as seen in figure 3: most of W, F, Corpus combinations have their 5% per-

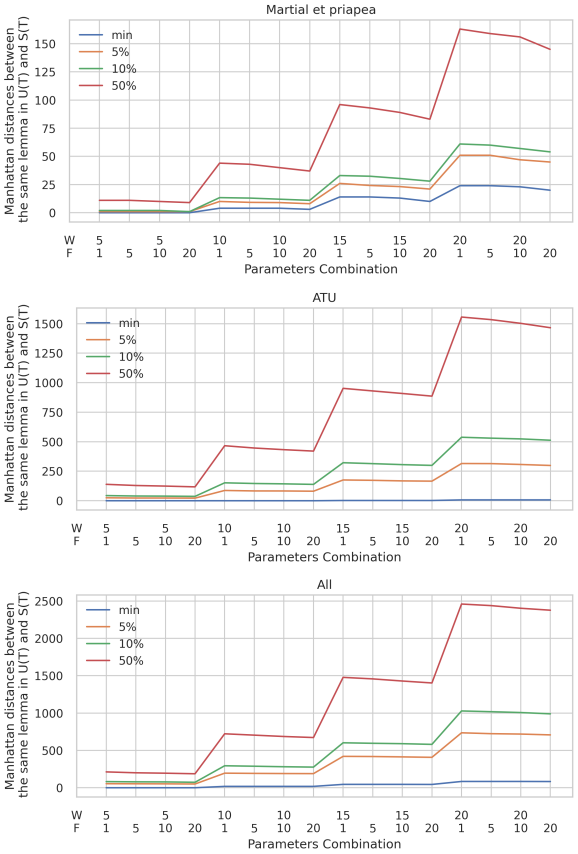

Figure 3: Variation of the distances based on W, F for each lemma's vector for the 1000 most frequent words in the full corpus, given three percentiles (5%, 10% and median) and the minimum value.

centile of distance which is not null. While each increase of W resulted in higher distances, the expected filtering effect of F also works on the noise: by ignoring less frequent co-occurring lemma, the increase of the value of F effectively lowers the distance, albeit very minimally. Indeed, F is the parameter that has the least impact on the computed distances.

3.2 On Classes

Based on this first observation, we want to evaluate what this noise can do to more advanced feature selections. To compare the effect on these major co-occurences' selection, we simply compare for each combination of parameters Word-set, W, F the set M of $U(T)$ with the set of $S(T)$. Given the differences of distances found in 3.1, this will show whether the noise accumulated through noisy windows is enough to influence the simple scoring provided by the Dice coefficient.

We first quantify the effect of a binary approach, *i.e.*, we check that $M_{S(T)}$ and $M_{U(T)}$ are equal.

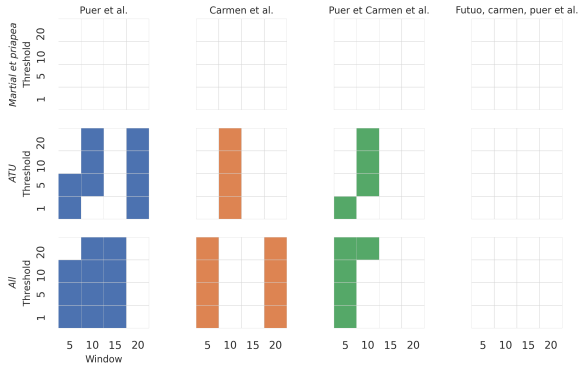

Figure 4: Binary matrix of $M_{S(T)} = M_{U(T)}$: colored cells mean the analysis on both versions of the corpus with the same parameters resulted in the same major co-occurrences.

Any situation where $M_{S(T)} \neq M_{U(T)}$ is a first proof that the absence of segmentation has an impact. In fact, with this approach, only 21.35% (41/192) of the run reaches perfect equality of their M for both versions of T, with varying results depending on the word-set and the corpus(*cf.* Figure 4):

- Overall, the word-set "Futuo et al." never results in fully similar M sets. This is probably due to very low frequencies of some of its members and would make the case, if such frequencies are acceptable from a statistical point of view, to very carefully segment the analyzed texts.
- As expected, any analysis using the corpus *Martial & Priapea* is heavily affected by its high amount of small ATU: none of them have similar output over its two versions. This corpus' size does not produce noise mitigation.
- Only the "Puer et al." regularly achieves equality over the corpus *ATU Corpus*, but it is not constant. This simply would advocate for segmentation of rich (S)ATU corpora as a prerequisite for analysis similar to the one we run here. It is also possible that the rather low frequency of "libellus" (546) in the *Carmen et al.* and *Puer et Carmen et al.* is responsible for some of the instability.
- Higher token counts do smooth the noise of features as the *All* corpora displays a higher stability between $M_{S(T)}$ and $M_{U(T)}$, specifically for *Puer et al.*, but M are still more often different than equal.

- The frequency threshold of co-occurrences F has a very small impact on major co-occurrences, while the window is irregularly affecting word sets: as an example, $W = 15$ never reaches equality for the combination *Carmen et al. + All* but it does for analysis *Puer et al.*; on the contrary, $W = 15$ fails on *Puer et al. + ATU Corpus*. This instability of the impact of W also advocates for relying on (S)ATU when doing such analysis.

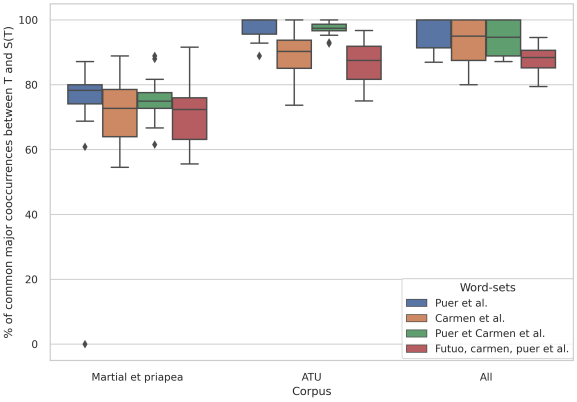

Figure 5: Dispersion of the overlap percentage or *retention rate* of M for each corpus and word-set combination. For *Puer et al.* on Martial et Priapea, major co-occurrences have a median similarity of below 80% over the 16 combinations of W, F.

For the relative rate and a more in-depth analysis of how classes vary from one version of the corpora to another, we propose to evaluate the *retention rate of classes* as a function of both sets M of major co-occurrences, computed:

$$1 - \frac{\left|M_{S(T)} - M_T\right| + \left|M_T - M_{S(T)}\right|}{\left|M_{S(T)} + M_T\right|}$$

With some exceptions on *Martial & Priapea* and *ATU Corpus*, the retention rate is generally over 60% for the smallest corpus, 80% for the two other[9] for a median number of major co-occurrences spanning from 8 (*Carmen et al.*) to 34 (*Futuo, carmen, puer et al.)* (*cf.* figure 6). However large the corpus and the word-set, the retention rate is globally high, particularly for the two biggest corpora, with one word-set being worse than the others (*Futuo et al.*).

Overall both metrics show an undisputed effect on major co-occurrences' selection. Corpus growth mitigates this effect as shown with the results of

[9]See appendix table 3

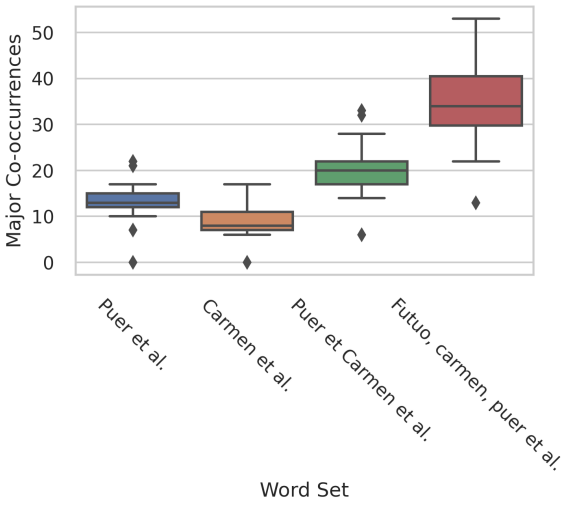

Figure 6: Number of Major Co-occurrences per Word Set

All vs. others, but it does not warranty equality of results between $S(T)$ and $U(T)$, as shown for $W = 20$ on *Puer et al.* or $W = 10$ and $W = 15$ for *Carmen et al.*. Carefully handling (S)ATU in relatively small corpora ($\leq 20M$ tokens) has to be an important step to strengthen any semantic statistical analysis.

3.3 On Features

Based on this second output, we want to identify if more advanced algorithms were as subject to variation with this normalized input. For each combination of W, F, which have the same clusters K such as $K_{S(T)} = K_T$ (*cf. figure 7*).

In this context, features have an impact that moves beyond the simple selection of classes, specifically for our first two corpora: on *Martial & Priapea*, for any word-set, there are no situations where all combinations of F, W provide the same clusters except for $k = 11$ and *Carmen et al.*, while none reaches the same clusters within the *ATU* corpus. In general, most combinations provide below 40% similar clustering for the first corpus and below 80% for the second. Similarly to the classes analysis, the size of the corpus mitigates the effect of $S(T)$ vs. $U(T)$: *All* has the biggest number of clusters which are equal in between both versions of the corpus. It, however, is still unstable and is unpredictable: while *Puer et al.* reaches 100% equality in clustering between $S(T)$ and $U(T)$ for $K = 5$ and $K = 10$, it falls down to 70% of similar results ariybd $K = 7$ and $K = 8$.

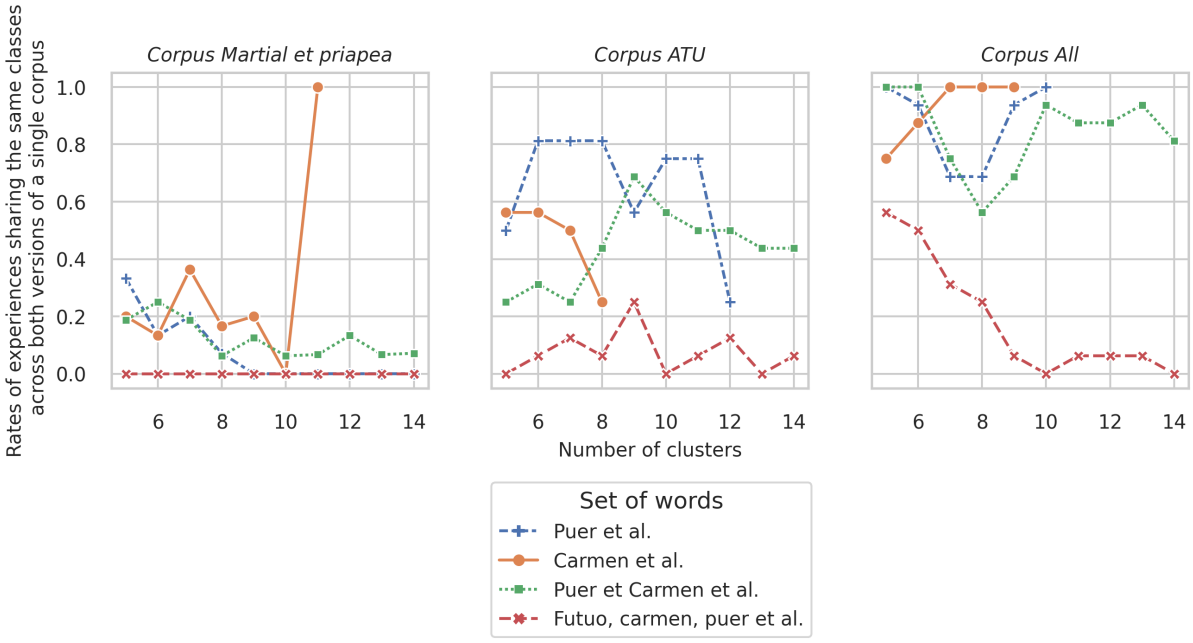

Figure 7: Ratio of experiments where $K_{S(T)} = K_T$ given the number of clusters for each corpus and word-set

4 Conclusion

When J. R. Firth used "company", which in our experiment becomes neighborhood and windows, what is meant is definitely more than having simply two words that follow each other: should two tweets in a timeline be treated as a single unit just because they appear in the same HTML "context"? The answer should be clear to any linguist, yet, texts and composite works have been treated this way by studies in DH or NLP[10].

In this experiment, we evaluated the impact of not taking into account the composition of texts. While working with statistics on such small corpora inherently requires caution, we demonstrated that treating digitized works as single cohesive units rather than as a patchwork of smaller units is altering the results of distributional analysis with corpora around 15 million tokens. In our experiments, only few combinations of the various parameters common to these applications (window size, frequency threshold, cluster size) yielded consistent results between an edited corpus of texts ($S(T)$) and a raw version of it ($U(T)$). With frequent words accross the corpus, with a small window ($W = 5$) and large corpora, the effect of noise is mitigated. But, if these words are more frequent in highly segmented works such as poetry compilation, the size of the overall corpus will have less impact on the issue.

These results do strengthen the necessity of metadata-enriched texts which allow for post-processing such as the one allowed by CapiTainS and the original XML TEI environment. It does definitely advocate for using declarations of segmentation with metadata such as CiteStructure (Cayless and Clérice, 2020) in order to make these corpora usable in machine actionable ways.

The current work was voluntarly limited in scope to both the Latin language and the use of deterministic methods. The Latin corpus is limited in size and does not provide a testing field for bigger corpora. Any experiment on larger corpus will have to deal with the annotation of larger corpora for segmentation purposes. The use of deterministic methods to represent clusters or distances between words allowed us for an easy and reproducible experiment. Applying the same approach to non-deterministic methods such as the one found in Word2Vec (Mikolov et al., 2013) and evaluating these results would provide a second testing field. However, the size of the corpus might already be a constraint difficult to overcome according to (Antoniak and Mimno, 2018).

Acknowledgments

We want to thank Florian Cafiero, Jean-Baptiste Camps, Marie Puren and Simon Gabay for their feedback on this article.

[10] See (Köntges, 2020) for example.

References

Maria Antoniak and David Mimno. 2018. Evaluating the Stability of Embedding-based Word Similarities. *Transactions of the Association for Computational Linguistics*, 6(0):107–119.

Jelke Bloem, Maria Chiara Parisi, Martin Reynaert, Yvette Oortwijn, and Arianna Betti. 2020. Distributional semantics for neo-Latin. In *Proceedings of LT4HALA 2020 - 1st Workshop on Language Technologies for Historical and Ancient Languages*, pages 84–93, Marseille, France. European Language Resources Association (ELRA).

Bruno Bureau. 2012. Quelques réflexions sur la notion de littérarité à partir de l'édition numérique de commentateurs anciens. *Interférences. Ars scribendi*, (6).

Luciano Canfora. 2016. *Conservazione e perdita dei classici*. Stilo editrice, Bari, Italie.

Catulle and Léon Herrmann. 1957. *Les deux livres de Catulle*. Latomus Revue d'Etudes Latines, Bruxelles (Berchem), Belgique. ISSN: 1378-8760.

Catulle and Georges Lafaye. 1932. *Poésies*. les Belles Lettres, Paris, France. ISSN: 0184-7155.

Hugh Cayless and Thibault Clérice. 2020. FR: Declarative Citation Structure · Issue #1957 · TEIC/TEI.

Thibault Clérice. 2017. Les outils CapiTainS, l'édition numérique et l'exploitation des textes. *Médiévales. Langues, Textes, Histoire*, 73(73):115–131.

Thibault Clérice. 2020a. Corpus latin antiquité et antiquité tardive lemmatisé.

Thibault Clérice. 2020b. Lasciva roma, priapea.

Thibault Clérice. 2020c. Pie extended, an extension for pie with pre-processing and post-processing.

Thibault Clérice. 2021a. Lasciva roma, additional texts.

Thibault Clérice. 2021b. Latin lasla model.

Thibault Clérice, Hippolyte Souvay, Etienne Ferrandi, Vincent Giovannangeli, Akim Ouchen, Léa Maronet, Émilien Arnaud, Krister Kruusmaa, and Jean Barré. 2021. lascivaroma/digiliblt: Release 0.0.64.

Gregory R. Crane. 2021a. Open greek and latin, corpus scriptorum ecclesiasticorum latinorum.

Gregory R. Crane. 2021b. Perseusdl/canonical-latinlit 0.0.752.

Maciej Eder. 2013a. Does size matter? Authorship attribution, small samples, big problem. *Digital Scholarship in the Humanities*, page fqt066.

Maciej Eder. 2013b. Mind your corpus: systematic errors in authorship attribution. *Literary and Linguistic Computing*, 28(4):603–614.

Stefan Evert. 2005. The statistics of word cooccurrences: word pairs and collocations.

John Rupert Firth. 1957. *Papers in linguistics, 1934-1951*. Oxford University Press.

Lynn S. Fotheringham. 2007. The Numbers in the Margins and the Structure of Cicero's "Pro Murena". *Greece & Rome*, 54(1):40–60. Publisher: Cambridge University Press.

Alain Guerreau. 1989. Pourquoi (et comment) l'historien doit-il compter les mots ? *Histoire & Mesure*, 4(1):81–105. Publisher: Persée - Portail des revues scientifiques en SHS.

Serge Heiden. 2010. The TXM Platform: Building Open-Source Textual Analysis Software Compatible with the TEI Encoding Scheme. In *24th Pacific Asia Conference on Language, Information and Computation*, volume 2/3, page 389-398, Sendai, Japan. Institute for Digital Enhancement of Cognitive Development, Waseda University.

Thomas Köntges. 2020. Measuring Philosophy in the First Thousand Years of Greek Literature. *Digital Classics Online*, pages 1–23.

Maurizio Lana. 2021. Metodologie e problematiche per una biblioteca digitale. il caso di digiliblt.

Enrique Manjavacas, Ákos Kádár, and Mike Kestemont. 2019. Improving lemmatization of non-standard languages with joint learning. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1493–1503, Minneapolis, Minnesota. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *arXiv:1301.3781 [cs]*. ArXiv: 1301.3781.

Joseph Morsel. 2015. Quelques propositions pour l'étude de la noblesse européenne à la fin du Moyen Âge. In *Discurso, memoria y representación : la nobleza peninsular en la Baja Edad Media (XLII Semana de Estudios Medievales de Estella)*, Discurso, memoria y representación : la nobleza peninsular en la Baja Edad Media (Actas de la XLII Semana de Estudios Medievales de Estella, 21 al 24 de julio 2015), pages 449–499, Estella, Spain. Gobierno de Navarra, Pamplona, Gobierno de Navarra, Pamplona.

Matthew Munson. 2017. *Biblical Semantics: Applying Digital Methods for Semantic Information Extraction to Current Problems in New Testament Studies*, 1 edition. Shaker, Aachen.

Nicolas Perreaux. 2012. Mesurer un système de représentation ? Approche statistique du champ lexical de l'eau dans la Patrologie Latine. In *Mesure et histoire médiévale*, pages 365–374, Tours, France. Publications de la Sorbonne.

Nicolas Perreaux and coraline rey. 2013. CBMA. Chartae Burgundiae Medii Aevi VII.“ Le “vocabulaire courant” en diplomatique : techniques et approches comparées ”. *Bulletin du Centre d'études médiévales d'Auxerre*, (17.1).

Martina Astric Roda, Philomen Probert, and Barbara McGillivray. 2019. Vector space models of Ancient Greek word meaning, and a case study on Homer. *Traitement Automatique des Langues*, 60(3/2019):63–87.

Christof Schöch. 2017. Topic Modeling Genre: An Exploration of French Classical and Enlightenment Drama. *Digital Humanities Quarterly*, 011(2).

Stéfan Sinclair and Geoffrey Rockwell. 2016. Voyant Tools.

Nathan Stringham and Mike Izbicki. 2020. Evaluating Word Embeddings on Low-Resource Languages. In *Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems*, pages 176–186, Online. Association for Computational Linguistics.

Radim Řehůřek, Petr Sojka, et al. 2011. Gensim—statistical semantics in python. *Retrieved from genism. org*.

A Online Resources

The code and output of this article can be found at https://github.com/lascivaroma/dont-worry-its-just-noise .

B Appendices

	Martial & priapea	ATU Corpus	All	Puer et al.	Carmen et al.	Puer et Carmen et al.	4 Fields
cunnus	33	42	43				✓
fello	11	14	28				✓
futuo	46	52	52				✓
irrumo	10	16	16				✓
lasciuus	35	155	300				✓
mentula	68	75	75				✓
paedico2	17	20	22				✓
carmen1	90	1753	3101		✓	✓	✓
lego2	95	3030	10252		✓	✓	✓
libellus	119	546	1190		✓	✓	✓
liber1	36	1773	21015		✓	✓	✓
poeta	53	1366	2944		✓	✓	✓
scribo	71	6282	20501		✓	✓	✓
dominus	112	7420	48519	✓		✓	✓
mater	43	2132	9271	✓		✓	✓
pater	73	5154	29927	✓		✓	✓
puella	120	989	2036	✓		✓	✓
puer	159	1812	5824	✓		✓	✓
uir	94	3908	20062	✓		✓	✓
uxor	63	1306	7832	✓		✓	✓

Table 2: Frequency distribution over the 3 corpora

Corpus	Word-Set	Experiments	Common classes mean	std	min	25%	50%	75%	max
Martial et priapea	Puer et al.	16.0	72.9	20.5	0.0	74.1	78.3	80.0	87.2
	Carmen et al.	15.0	71.8	11.6	54.5	64.0	72.7	78.6	88.9
	Puer et Carmen et al.	16.0	75.7	6.9	61.5	72.7	74.9	77.6	88.9
	Futuo, carmen, puer et al.	16.0	71.3	9.7	55.6	63.2	72.4	75.9	91.6
ATU	Puer et al.	16.0	96.9	4.4	88.9	95.6	100.0	100.0	100.0
	Carmen et al.	16.0	88.6	9.8	73.7	85.1	90.3	93.8	100.0
	Puer et Carmen et al.	16.0	97.3	2.3	92.7	96.7	97.4	98.7	100.0
	Futuo, carmen, puer et al.	16.0	86.6	6.7	75.0	81.7	87.5	91.9	96.7
All	Puer et al.	16.0	96.3	5.8	87.0	91.4	100.0	100.0	100.0
	Carmen et al.	16.0	93.5	7.1	80.0	87.5	95.0	100.0	100.0
	Puer et Carmen et al.	16.0	94.3	5.1	87.2	88.9	94.7	100.0	100.0
	Futuo, carmen, puer et al.	16.0	87.9	4.4	79.5	85.2	88.4	90.6	94.5

Table 3: Ratio of common classes over experiments

NLP in the DH pipeline: Transfer-learning to a Chronolect

Aynat Rubinstein
The Hebrew University
of Jerusalem
aynat.rubinstein@mail.huji.ac.il

Avi Shmidman
Bar-Ilan University
Dicta: The Israel Center for Text Analysis
avi.shmidman@biu.ac.il

Abstract

A big unknown in Digital Humanities (DH) projects that seek to analyze previously untouched corpora is the question of how to adapt existing Natural Language Processing (NLP) resources to the specific nature of the target corpus. In this paper, we study the case of Emergent Modern Hebrew (EMH), an under-resourced chronolect of the Hebrew language. The resource we seek to adapt, a diacritizer, exists for both earlier and later chronolects of the language. Given a small annotated corpus of our target chronolect, we demonstrate that applying transfer-learning from either of the chronolects is preferable to training a new model from scratch. Furthermore, we consider just how much annotated data is necessary. For our task, we find that even a minimal corpus of 50K tokens provides a noticeable gain in accuracy. At the same time, we also evaluate accuracy at three additional increments, in order to quantify the gains that can be expected by investing in a larger annotated corpus.

1 Introduction

Digital Humanities (DH) projects that deal with understudied languages, including many historical languages, often face a "throughput bottleneck" due to the lack of Natural Language Processing (NLP) resources trained for the specific language variety they document. In many cases, there is a closely related language, typically a modern standard variety, for which there exist large digital corpora, goldsets manually-annotated for various features, and NLP tools (e.g., morphological analyzers, syntactic parsers). It is an open question to what extent it is useful to adapt tools tailored for one *chronolect* (a language used in a particular point in time), in order to build resources for a closely related yet different chronolect (Baron and Rayson 2008; Schneider 2020 on spelling normalization for English chronolects; Pettersson et al. 2013 for Early Modern Swedish spelling, tagging, and parsing; Pettersson and Nivre 2015 for verb phrase extraction in Early Modern Swedish; Hana et al. 2011 for morphological tagging of Old Czech).

The researcher is faced in such a case with questions such as: how much data is needed for training? Is it better to fine-tune existing NLP models, which were trained to fit a different chronolect, or instead to train new models only on the chronolect of interest? These are the questions that arose in the framework of a DH project "The Jerusalem Corpus of Emergent Modern Hebrew" (JEMH; Rubinstein 2019), which aims to digitize and enable linguistic analysis of multi-genre archival material in Emergent Modern Hebrew (EMH; Hebrew of the early 20th century). Diacritization of this corpus is a critical step both in order to increase its accessibility (in particular because the diacritics are a necessary prerequisite for the application of text-to-speech algorithms), and also for downstream NLP tasks such as intelligent search, relation extraction, and linguistic analysis of the historical materials.

Although EMH is a severely undersourced Hebrew chronolect, diacritization models do exist for other Hebrew chronolects. We thus aim to answer the following questions: how much diacritized EMH data must be assembled, and how can we best leverage existing models from other Hebrew chronolects? We manually add annotations of diacritics to a representative sample of the JEMH Corpus and use this new goldset to train a neural-net diacritizer for EMH. The resulting diacritizer is the first NLP tool developed for this historical langauge variety. It can be incorporated into the pipeline of any DH project dealing with EMH, providing valuable disambiguations of the text.

We compare two approaches to the creation of the diacritizer for EMH. The *indirect* approach uses the EMH goldset to fine-tune existing dia-

Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH), pages 106–110
December 19, 2021.

critizers on top of two well-resourced chronolects of Hebrew which are closely-related to our target chronolect (one variety predates EMH and one is the present-day standard variety). In the *direct* route, we train a new model for EMH using just our (small) annotated goldset. We provide, for each approach, an estimate of the lower bound on the amount of training data needed to reach acceptable performance, discussing the implications of our findings for planning DH projects.

2 Linguistic challenges

The use of Hebrew as a spoken language only began around the turn of the 20th century, and the norms of Modern Hebrew - lexical, morphological, syntactic, and orthographical - only crystallized during the decades afterward (Rosén, 1956; Blanc, 1968; Ben-Hayyim, 1992; Rabin, 1999; Reshef, 2013, 2015, 2016; Doron et al., 2019b). However, knowledge about previous stages of the language was accessible to learners through canonized texts which formed the basis of Jewish education throughout history (Doron et al. 2019a). Therefore, EMH constitutes a chronolect that is considerably distant from Modern Hebrew. First of all, on the lexical plane, neologisms for many everyday objects such as "kitchen" or "newspapers" were only just being invented for the first time, and were often referred to by multiword circumlocutions instead (e.g., for kitchen: בֵּית הַבִּשּׁוּל *bet habišul*, lit. 'house of cooking', rather than the modern word מִטְבָּח *mitbaħ*, or, for newspaper, מִכְתָּב עִתִּי *mixtav ʕiti*, lit. 'periodical letter', rather than עִתּוֹן *ʕiton*). On the morphological plane, we find much use of nominal patterns now completely obsolete (e.g., מְפֹאָרָה *mefoʔarah* 'magificent', rather than מְפֹאֶרֶת *mefoʔeret*, or מַתְחֶלֶת *matħelet* 'start (participle, fem.)' rather than מַתְחִילָה *matħilah*). Finally, and most significantly, plene orthography - now normative in Modern Hebrew - had not yet been embraced, causing ambiguities to abound. As has been noted, Modern Hebrew by itself is already highly ambiguous morphologically (Wintner, 1998; Tsarfaty et al., 2019); however, without the norms of plene spelling, the ambiguity is amplified considerably. For example, in EMH חדשים *ħdšim* may be analyzed as one of three words: חֲדָשִׁים *ħadašim* 'new (pl.)', חֳדָשִׁים *ħodašim* 'months', or חָדְשַׁיִם *ħodšayim* 'two months', whereas in Modern Hebrew, each is represented by a distinct unambiguous spelling, via the addition of matres lectionis: חדשים *ħdšim*, חודשים *ħodšim*, or חודשיים *ħodšiim*.

Resolving these challenges is crucial for natural lanuage processing of the formative and highly influential EMH corpus of Hebrew. Adding diacritics can dramatically reduce ambiguity and is the tool we sought to develop.

3 Diacritizaton

Modern Hebrew (similar to other Semitic languages such as Arabic and Syriac) is generally written and published with many of the vowels omitted. In EMH, even fewer vocalizations are marked. *Diacritization* refers to the specification of all vowels as part of the written word. The set of diacritics generally used in Hebrew is termed "Tiberian diacritization", and consists of a set of a dozen essential marks placed below, within, or above the characters (Golinets 2013). Letters can be optionally geminated (marked with a dot in the center of the letter), leading to a total of 24 possible diacritic permutations for each letter.

A given non-diacritized Hebrew word generally admits to multiple possible diacritizations, each representing a different semantic and morphological analysis of the word. Thus, diacritization cannot be automated via a simple lookup table; rather, it is necessary to use contextual information to choose from among the multiple analyses. Several machine-learning systems have been developed to perform this task (Choueka and Neeman, 1995; Gal, 2002; Gershuni and Pinter, 2021). The current state-of-the-art for Modern Hebrew is the LSTM-based diacritizer developed by Dicta (Shmidman et al., 2020), as per external evaluations performed by Gershuni and Pinter (2021).

However, as noted, the EMH chronolect presents a new set of challenges, and, indeed, automated diacritization systems for Modern Hebrew falter on EMH. We therefore set out to determine how large a corpus would be necessary to train the same kind of LSTM specifically for EMH, and to determine whether it would be beneficial, alternatively, to attempt a transfer-learning architecture based upon the pre-trained Modern Hebrew LSTM model.

4 Experiments

Data Our data consists of a selection of texts from the JEMH Corpus (Rubinstein, 2019) to which diacritics have been added manually by experts. For manual annotation, we used the *Nakdan*

	Words	Years
Literature	129,453	1858-1932
Ephemera	14,993	1862-1941
Total	144,446	

Table 1: Goldset of EMH with manually-annotated diacritics

Pro interface by Dicta.[1] Most of the corpus represents a literary genre,[2] complemented by ephemera from the JEMH Street Ads supcorpus. Table 1 provides information about the size of the annotated corpus and the period it spans. We release this vocalized corpus to the public domain.[3]

Implementation and Results We divided our corpus of diacritized EMH literature into 120K words for a training set and 11K words for the test set. In order to track the effect of the size of the corpus, we train four separate LSTM models, using four subsets of the training corpus, of sizes 50K, 75K, 100K, and 120K. For each subset, we train two models: (1) We train the LSTM from scratch, using only the vocalized data in the training subset. We train for 100 epochs. (2) We adopt a transfer-learning approach: we start with Dicta's Modern Hebrew LSTM model, which was trained on over 2 million words of Modern Hebrew text. We then fine-tune this model for 100 additional epochs, using only the data in the EMH training subset. Regardless of the subset used for training, we always use the same test set, to ensure consistency.

In evaluating the accuracy of the resulting model, we separately consider two approaches: (1) We test the ability of the LSTM model to predict the correct vocalization without any constraints. For each word, we run a beam search across the top eight vocalization predictions for each letter, and we take the top scoring beam. (2) We test the ability of the LSTM model to choose the correct vocalization option from a set of options provided by a wordlist.[4] Thus, as opposed to the previous approach, here we constrain the LSTM to known valid vocalization options. For each word, we calculate the LSTM score for each of the vocalization options in the wordlist, and we take the top scoring option.[5] Results are shown in Table 2.[6]

Training Size	New Train		Fine-tune	
	LSTM	LSTM +Wordlist	LSTM	LSTM +Wordlist
0	—	—	78.30%	**84.57%**
50,000	70.47%	81.92%	78.60%	**85.86%**
75,000	73.39%	83.38%	80.20%	**86.80%**
100,000	76.14%	84.44%	80.92%	**87.06%**
120,000	77.29%	85.15%	81.98%	**87.38%**

Table 2: We test how much training data is necessary to train an LSTM model to diacritize the EMH chronolect. First (col 2-3) we show the effects of training a new model from scratch based on the specified number of tokens, with and without the use of a wordlist restricting choices to known valid forms. Next (col 4-5) we show the superior effects of transfer-learning from an existing robust model for Modern Hebrew. The initial row shows the performance of the existing model on the EMH test corpus. We then show the improvement gained by fine-tuning this model with increasing sizes of EMH texts.

Training Size	New Train		Fine-tune	
	LSTM	LSTM +Wordlist	LSTM	LSTM +Wordlist
0	—	—	80.66%	90.19%
120,000	64.98%	83.39%	74.01%	88.09%

Table 3: We evaluate an out-of-domain text (ephemera) within the target chronolect. Retraining shows no benefit at all, even with the entire 120K word corpus. Whether we retrain from scratch or fine-tune on top of the existing Modern Hebrew model, we find that the training from the EMH literary corpus only reduces the accuracy. With ephemera, it is preferable to stick with the existing Modern Hebrew model.

Next, we test the effect of the training corpus on an out-of-domain corpus within the chronolect. Whereas the previous experiment involved training and test corpora both drawn from literary EMH, here we keep the same literary training corpus, but we use a different genre of EMH (ephemera) as the test corpus. Results are shown in Table 3.

Finally, we examine the effects of doing the transfer-learning from an *earlier* chronolect, rather than from a later chronolect. In the previous experiments, we took a pre-trained model from Modern Hebrew, and we fine-tuning it for the EMH

[1] https://nakdanpro.dicta.org.il.

[2] Obtained from the *Ben-Yehuda Project* (https://benyehuda.org/) snapshot in 2014.

[3] https://github.com/JEMHcorpus/corpora/tree/master/diacritized.

[4] We use a high-coverage Hebrew lexicon curated in-house at Dicta. For details regarding the lexicon, see (Shmidman et al., 2020), 199. We further augmented the wordlist for this project by adding support for morphological expansions typical of EMH, such as those described in section 2.

[5] If the word is not in our wordlist at all, then we default to the top LSTM beam-search prediction, as in the first approach; within our EMH corpus, this situation occurs regarding a small minority of cases, approximately 1.5% of the corpus.

[6] Percentages displayed here (and in all other tables as well) reflect word-level accuracy. For a given word in the text, we consider the prediction correct if and only if all the diacritic marks on the word are correct, including proper gemination and selection of the 'shin' dot, and including removal of all matres lectionis. Note that punctuation and non-Hebrew words within the text are not included in this calculation (because their inclusion would artificially inflate the score).

Training Size	Fine-tune over Rabbinic Hebrew		Fine-tune over Modern Hebrew	
	LSTM	LSTM +Wordlist	LSTM	LSTM +Wordlist
0	70.80%	80.03%	78.30%	**84.57%**
50,000	76.26%	83.89%	78.60%	**85.86%**
75,000	77.23%	84.04%	80.20%	**86.80%**
100,000	78.69%	84.91%	80.92%	**87.06%**
120,000	79.34%	85.39%	81.98%	**87.38%**

Table 4: We introduce the question of how transfer-learning from an earlier choronolect would differ from the case of a later choronolect. In columns 2 and 3, we evaluate the effect of fine-tuning on top of Rabbinic Hebrew, a Hebrew chronolect of the middle ages and the subsequent few centuries. In columns 4 and 5, we show the comparison to the results from Table 2 of fine-tuning on top of Modern Hebrew. As we can see here, from the outset, the pre-trained model of Modern Hebrew performs substantially better on the EMH corpus than the pre-trained model of Rabbinic Hebrew. Nevertheless, fine-tuning on top of the earlier Rabbinic Hebrew corpus does provide substantial results. Enlarging the training corpus progressively closes the gap between the Rabbinic and Modern models.

chronolect; essentially, we were testing what it would take to retroject the Modern Hebrew model to the Hebrew of some 100 years prior. However, the Dicta Nakdan also contains a robust model for Rabbinic Hebrew, a dialect of Hebrew found in abundance in Jewish legal texts from the middle ages, and the first few centuries afterward. Fine-tuning this model would be testing a transfer of the opposite direction: we evaluate what it would take to adapt the medieval and post-medieval Rabbinic model several centuries forward. We fine-tune the Rabbinic model with the same four subsets with which we fine-tuned the modern model, and we compare the results in Table 4.

5 Discussion

The results of these experiments lead us to the following observations:

First of all, in all cases, transfer-learning from a different Hebrew chronolect is better than training from scratch. This is true whether the source chronolect is later or earlier. At the same time, fine-tuning on top of a later chronolect was clearly superior to fine-tuning on top of an earlier chronolect, likely because a later chronolect will still have remnants of the earlier chronolect, while the same cannot be said of an earlier choronolect.

Additionally, in all cases, the wordlist filter improved scores considerably. Without the wordlist filter, the LSTM is free to choose any diacritization sequence at all, and in many cases ends up predicting a sequence that is not a valid word. Although many diacritization patterns are limited to certain letter configurations, and we expect the LSTM to learn these patterns, in other cases the choice of one pattern or another is fairly arbitrary, and just indicates natural development stages of the language. Thus, the wordlist provides the necessary knowledge to ensure that the predicted diacritics are indeed a pattern known to apply to the given word.

Regarding the question of how much training data is necessary: Of course, as expected, the larger the training corpus, the more accurate the result. However, the tables indicate just how much of an improvement we can expect with every additional 20,000-25,000 tokens. As we see, there is generally an increase of somewhere between half a percent and one and a half percent from stage to stage. Presumably, if we were to continue to enlarge our training corpus, the accuracy would continue to rise, until it reached a point of diminishing returns.

Importantly, although the full 120K training corpus yields best results, even a 50K corpus succeeds in providing a palpable improvement when fine-tuning on top of other chronolects. This suggests that investing in a small annotated corpus is a viable route for DH projects that seek to adapt NLP tools from neighboring chronolects.

Finally, the successful effects of the transfer-learning are limited to texts whose genre is represented within the training corpus. In contrast, when we tested our fine-tuned models on the out-of-domain ephemera corpus, we found that the transfer-learning actually lowered the accuracy of the model. This result may be related to the particular languages we tested, however, showing that the language of the EMH ephemera is closer to the present-day language than other EMH materials.

6 Conclusion

This paper examines a central concern in DH: the question of how much data is needed to train viable NLP tools for under-resourced chronolects. We addressed these questions for the specific case of a historical dialect of Hebrew, showing an advantage for fine-tuning a diacritization model using a small annotated corpus (i.e., transfer-learning) over direct training of a new model for the chronolect of interest. Our findings have implications for any project that aims to adapt an NLP algorithm to an alternate chronolect. Given access to existing NLP tools, it is likely that producing even a small annotated cor-

pus of historical materials from the relevant time period will result in substantial gains.

Acknowledgments

We acknowledge the substantial help of our programmer Cheyn Shmuel Shmidman, and of our linguistic experts, Binyamin Ehrlich and Tsion Eliash, who were responsible for the creation of the diacritized goldset of EMH texts.

This research was supported by the Israel Science Foundation (grant no. 2299/19) and by Dicta: The Israel Center for Text Analysis, headed by Prof. Moshe Koppel.

References

Alistair Baron and Paul Rayson. 2008. VARD2: a tool for dealing with spelling variation in historical corpora. Postgraduate Conference in Corpus Linguistics ; Conference date: 22-05-2008.

Ze'ev Ben-Hayyim. 1992. *The struggle for a language*. The Academy of the Hebrew Language, Jerusalem. In Hebrew.

Haim Blanc. 1968. The Israeli *koine* as an emergent national standard. In Joshua A. Fishman, editor, *Language problems of developing nations*, pages 237–251. Wiley, New York.

Yaacov Choueka and Yoni Neeman. 1995. Nakdan-text,(an in-context text-vocalizer for modern hebrew). In *BISFAI-95, The Fifth Bar Ilan Symposium for Artificial Intelligence*.

Edit Doron, Malka Rappaport Hovav, Yael Reshef, and Moshe Taube. 2019a. Introduction. In Edit Doron, Malka Rappaport Hovav, Yael Reshef, and Moshe Taube, editors, *Language Contact, Continuity and Change in the Genesis of Modern Hebrew*, pages 1–31. John Benjamins, Amsterdam.

Edit Doron, Malka Rappaport Hovav, Yael Reshef, and Moshe Taube, editors. 2019b. *Language Contact, Continuity and Change in the Genesis of Modern Hebrew*. John Benjamins, Amsterdam.

Ya'akov Gal. 2002. An hmm approach to vowel restoration in arabic and hebrew. In *Proceedings of the ACL-02 workshop on Computational approaches to semitic languages*, pages 1–7. Association for Computational Linguistics.

Elazar Gershuni and Yuval Pinter. 2021. Restoring hebrew diacritics without a dictionary.

Viktor Golinets. 2013. Masora, Tiberian. In Geoffrey Khan, editor, *Encyclopedia of Hebrew Language and Linguistics*. Brill, Leiden.

Jirka Hana, Anna Feldman, and Katsiaryna Aharodnik. 2011. A low-budget tagger for old Czech. In *Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, pages 10–18, Portland, OR, USA. Association for Computational Linguistics.

Eva Pettersson, Beáta B. Megyesi, and Jörg Tiedemann. 2013. An smt approach to automatic annotation of historical text. In *Proceedings of the Workshop on Computational Historical Linguistics at NODALIDA*, volume 87 of *NEALT Proceedings Series 18*, pages 54–69, Vilnius, Lithuania. Linköping University Electronic Press, Sweden.

Eva Pettersson and Joakim Nivre. 2015. Improving verb phrase extraction from historical text by use of verb valency frames. In *Proceedings of the 20th Nordic Conference of Computational Linguistics (NODALIDA 2015)*, pages 153–161, Vilnius, Lithuania. Linköping University Electronic Press, Sweden.

Chaim Rabin. 1999. What was the revival of the Hebrew language? *Linguistic Studies*, pages 359–376. In Hebrew.

Yael Reshef. 2013. Revival of Hebrew: Grammatical structure and lexicon. In Geoffrey Khan, editor, *Encyclopedia of Hebrew Language and Linguistics*, volume 3, pages 397–405. Brill, Leiden.

Yael Reshef. 2015. *Hebrew in the Mandate period*. The Academy of the Hebrew Language, Jerusalem. In Hebrew.

Yael Reshef. 2016. Written Hebrew of the revival generation as a distinct phase in the evolution of Modern Hebrew. *Journal of Semitic Studies*, 61(1):187–213.

Haiim Rosén. 1956. *Ha-'ivrit šelanu*. Am Oved, Tel Aviv. In Hebrew.

Aynat Rubinstein. 2019. Historical corpora meet the digital humanities: the Jerusalem Corpus of Emergent Modern Hebrew. *Language Resources and Evaluation*, 53(4):807–835.

Gerold Schneider. 2020. Spelling normalisation of Late Modern English. In Merja Kytö and Erik Smitterberg, editors, *Late Modern English: Novel encounters*, pages 244—-268. John Benjamins, Amsterdam.

Avi Shmidman, Shaltiel Shmidman, Moshe Koppel, and Yoav Goldberg. 2020. Nakdan: Professional Hebrew diacritizer. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 197–203, Online. Association for Computational Linguistics.

Reut Tsarfaty, Amit Seker, Shoval Sadde, and Stav Klein. 2019. What's wrong with Hebrew NLP? and how to make it right. *CoRR*, abs/1908.05453.

Shuly Wintner. 1998. Towards a linguistically motivated computational grammar for Hebrew. In *Computational Approaches to Semitic Languages*.

Using Computational Grounded Theory to Understand Tutors' Experiences in the Gig Economy

Lama Alqazlan[1], Rob Procter[1,3] and Michael Castelle[2,3]
[1]Department of Computer Science, University of Warwick
[2]Centre for Interdisciplinary Methodologies, University of Warwick
[3]The Alan Turing Institute, London, UK
{Lama.alqazlan|Rob.Procter|M.Castelle.1}@warwick.ac.uk

Abstract

The introduction of online marketplace platforms has led to the advent of new forms of flexible, on-demand (or 'gig') work. Yet, most prior research concerning the experience of gig workers examines delivery or crowdsourcing platforms, while the experience of the large numbers of workers who undertake educational labour in the form of tutoring gigs remains understudied. To address this, we use a computational grounded theory approach to analyse tutors' discussions on Reddit. This approach consists of three phases including data exploration, modelling and human-centred interpretation. We use both validation and human evaluation to increase the trustworthiness and reliability of the computational methods. This paper is a work in progress and reports on the first of the three phases of this approach.

1 Introduction

The introduction of online platforms has contributed significantly to the changing economic structure and nature of employment (Kenney and Zysman, 2016). In labour markets, platforms are utilised to match and mediate between independent workers and consumers through flexible arrangements, where workers are contracted to perform single discrete tasks and are paid upon task completion, known as the "gig economy" (Broughton et al., 2018; Koutsimpogiorgos et al., 2020). This can provide employment opportunities for those who struggle to find work and the underemployed to supplement their income (Clark, 2021). Conversely, gig employment is also known to be chronically precarious and associated with low remuneration (Duggan et al., 2021; Edward, 2020). To achieve their earnings goals, gig workers have been found to work on average 71 hours, compared to the standard 45 hours per week (Heeks et al., 2021). These hours, however, are not fully paid, as workers spend considerable time waiting on or looking for gigs to perform (Anwar and Graham, 2021). Moreover, in countries such as the UK, gig workers are not eligible for minimum wage, overtime, sick or holiday pay and health insurance primarily because they are classified as "independent contractors" and not employees (Clark, 2021; Heeks et al., 2021), which is currently the focus of legal debates. While some platforms act only as a channel connecting gig workers with customers, others are heavily involved in the process — from job assignments and pricing to work assessment through timing and reviews (Koutsimpogiorgos et al., 2020). This raises doubts as to whether gig workers have sufficient job autonomy to be considered "independent contractors" or if gig economy companies are taking advantage of the current binary worker classification system (i.e., employed or self-employed) to avoid providing employee benefits to employee-like workers (Clark, 2021). Furthermore, the platform economy raises ethical issues relating to their reliance on algorithmic management (Jarrahi et al., 2021; Tan et al., 2021), the lack of transparency on their operation (Jarrahi and Sutherland, 2019), and their inherent power asymmetries, as gig workers are seen as the less powerful party in the process (Koutsimpogiorgos et al., 2020). Moreover, working on some platforms could negatively impact workers' career development, social capital and networks (Duggan et al., 2021).

The issues relating to platform labour are varied and may be more obvious on some platforms than others. Some argue that location-based platforms are less flexible and autonomous than location-independent platforms (Woodcock and Graham, 2019). Working on "microtask" platforms can limit workers' skills and career development (Rani and Furrer, 2019), while workers on "macrotask" platforms feel that their work is underappreciated and underpaid (Nemkova et al., 2019). Many of these issues are especially true for people who rely

Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH), pages 111–120
December 19, 2021.

entirely on the gig economy for their livelihood (Glavin et al., 2021; Koutsimpogiorgos et al., 2020).

Most prior research concerning the experience of gig workers examines platforms such as Uber, Deliveroo and MTurk (Cano et al., 2021; Howcroft and Bergvall-Kåreborn, 2019), while the experiences of the large number of educational labourers who perform tutoring gigs remain understudied. Tutors' experiences are distinct from other gig workers as their tasks typically involve teaching one-to-one sessions which have unique challenges, as these are held in real-time and are of a reasonable length. Therefore, this study aims to contribute to this growing area of research by exploring the experiences of tutors in the gig economy, the problems they face and how their experiences compare to those of other types of gig workers.

2 Methodology

2.1 Data

The discussion forum and social news aggregator Reddit was manually examined to find relevant subreddits using keywords and platform names, resulting in eighteen subreddits (see Table 1). To retrieve related discussions, we used the Reddit API Wrapper (PRAW), which resulted in approximately 52,000 posts and comments. Then, preprocessing tasks were conducted to convert Reddit's free text into a structure amenable to text mining. This included the removal of stopwords, punctuation, emojis, URLs, lowercasing, lemmatization, and tokenization, resulting in a vocabulary size of 7,491. Samples of posts after the preprocessing steps are shown in Table 2. Finally, before collecting the data, ethics approval was obtained from the Biomedical and Scientific Research Ethics Committee at the University of Warwick.

Platform-specific Subreddits	Subject-specific and other related Subreddits
Vipkid, Preply, MagicEars, Qkids, Cambly, DaDaABC, iTalki, iTutor, Gogokid, Tombac, ZebraEnglish, GoGoKidTeach, Palfish.	OnlineESLTeaching, online_tefl, teachingonline, onlineESLjobs, TeachEnglishOnline.

Table 1. List of subreddits included in the study

2.2 The model

The methodology consists of three main phases.

Preprocessed tokenized text
['add','hour','available','asian','company','peak','hour', 'around','west','coast','usa']
['advice','try','engage','response','move','fair','student', 'learn','time','get','waste']
['best','case','scenario','mean','lot','new','kid','flock','online' , 'long', 'lesson', 'hope', 'put', 'soon', 'right', everyone', 'spiral']

Table 2. Samples of posts after data preprocessing steps

2.2.1 Phase One: Data Exploration

(a) Initial data exploration

Thorough initial data exploration is essential to gain early data-driven insights and to avoid the blind use of unsupervised machine learning algorithms. This is especially important if these are used to replace human reading and judgments on large-scale data, as models output may be varied, misleading or even wrong (DiMaggio, 2015; Grimmer and Stewart, 2013). Thus, we used a constructivist grounded theory (GT) procedure (Charmaz, 2006) to analyse a subset of the data: we randomly selected two small subreddits (GoGoKidTeach and Palfish) with about 160 posts and comments, resulting in a list of themes to be utilised thereafter. GT analysis, however, has its limitations. First, it relies on a series of judgments while coding and interpreting the data, potentially bringing subjectivity into the analysis. Second, the manual reading and analysis of large datasets is onerous and time consuming. These were countered in the next step.

(b) Validation and further exploration

To ensure the validity and reliability of the themes identified by GT and expand the data exploration to include the whole corpus, we used Latent Dirichlet Allocation (LDA) (Blei et al., 2003) topic modelling (TM). Labels were produced to describe each topic by examining the 5 highest weighted documents, as close reading of the associated documents can provide sufficient understanding of each topic's essence (Brookes and McEnery, 2019). Topic labels were then compared against GT themes. To assess the validity and reliability of the analysis, one can compare the themes and the topics which emerge from both GT and LDA, a method that Boussalis and Coan (2016) refer to as *concurrent validation*. Furthermore, LDA, with its scalability, accelerates the analysis and may uncover additional topics due to the increased dataset size. The similarity in these two methods

allows for validation as both techniques are exploratory, data-driven, iterative and test intermediate versions. While the distinctions between them help to enhance scalability, minimize subjectivity, and reduce the time required for analysis (Baumer et al., 2017).

(c) Terms extraction

In this step, we collect the terms for each topic required for Phase Two. First, for topics identified in both GT and LDA or LDA only, the 20 highest weighted terms for each topic in LDA were selected, noting that terms that were judged not to be directly related to the topic were excluded. Second, a list of terms was proposed for topics that appeared only in GT.

2.2.2 Phase Two: Modelling — Query-driven topic modelling

In Phase One, the topics of discussions were identified, confirmed, and expanded. Subsequently, query-driven topic modelling (QDTM) (Fang et al., 2021) will be used to model topics from the whole corpus, where the input for the model is a set of query terms for each topic.

The functions and advantages of using this approach are threefold: (1) QDTM can model topics that LDA failed to detect, as LDA ascertains topics using the frequency with which words appear together and thus infrequent topics, regardless of their importance, may not be detected; (2) QDTM uses term expansion techniques, where the input queries are expanded to a set of concept terms using one of three approaches: frequency-based extraction, KL-divergence based extraction, and relevance modelling with word embeddings. This step is particularly helpful, as for most of the topics depend only on terms automatically generated by LDA (with no terms were added to these topics to avoid bias); and (3) QDTM feeds these concept terms into a two-phase framework based on a variation of a Hierarchical Dirichlet Process (HDP) to form the main topic and subtopics. This way of structuring topics is superior to traditional TM (i.e., LDA) which only mines monolayer topics and fails to discover the hierarchical relationship among topics and so is considered an effective way to organise and navigate large-scale data (Dumais and Chen, 2000; Johnson, 1967). This helps to guide analysis and provide the desired level of detail for Phase Three. Finally, the aim is to conduct a human evaluation of topic coherence and exclude non-useful topics.

2.2.3 Phase Three: Human-Centred Interpretation — Computational Grounded Theory (CGT)

By this stage, a list of computationally identified, confirmed and human-evaluated topics will have been obtained, each represented by their highest-weighted documents. To perform CGT, samples of the documents will be read in detail and analysed using the conventional GT process. This will add interpretation to the analysis to better understand the topics and assist in the development of higher-level conclusions. The use of TM was essential to discover and classify topics, as the data is too large to be manually read and analysed accurately and efficiently. It also helped to reduce the subjectivity that may come from detailed reading in traditional data analysis methods as a researcher may assign more weight to topics that corroborate their pre-held opinions (Morse, 2015; Nelson, 2017). Finally, as GT involves moving back and forth between the results of the analysis and the data, CGT will take a similar approach as the researcher will return to the data via a structured qualitative analysis after having identified and confirmed the topics (Nelson, 2017).

3 Results

3.1 GT Analysis

The final high-level themes from Phase One step **(a)** Initial data exploration are listed in Table 3. See Appendix A for a description of each theme.

Theme no.	Label
Theme 1	Hiring process
Theme 2	New contracts
Theme 3	Job requirements
Theme 4	How tutors consider the job
Theme 5	The class and the students
Theme 6	Teaching material and methods
Theme 7	Bookings and working hours
Theme 8	Payment
Theme 9	Rating system
Theme 10	Reasons to join or leave a platform
Theme 11	COVID-19-related discussions
Theme 12	Technical problems
Theme 13	Miscommunication with platform management
Theme 14*	Expressing feelings and sharing experiences
Theme 15*	Seeking and providing help and advice

Table 3. GT themes

***Note:** Themes 14 and 15, which reflect the underlying purpose of the posts, are excluded from the comparison with LDA as they were generated from observation of the data and due to their abstract nature LDA is not expected to model them.

3.2 TM Results

This section presents the results of Phase One step **(b) Validation and further exploration**. This includes the process of finding the optimal LDA

model and the final results of the two models in terms of comparing topic labels to GT themes.

At first, the number of topics, K, was assigned the value 13 (see Appendix B, Table B1). Comparing topics labels to GT results, the model found nine topics that correlated with GT themes and one new topic regarding bank transfers and transaction fees. Nonetheless, topics were missing compared to the GT themes, namely: a) reasons to join or leave a platform, b) COVID-19 related discussions, c) teaching material and methods, and d) miscommunication with platform management. Therefore, increasing the number of topics should improve our model. However, since the process of iteratively changing the number of topics and evaluating the results can be time-consuming and impractical, we used the *Tmtoolkit* Python package to compute and evaluate several models in parallel using state-of-the-art theoretical approaches, with the topic range set to between 5 and 30 (see Figure 1).

Here, the optimal number of topics is the one that minimises the average cosine distance between every pair of topics (Cao et al., 2009), and has minimal divergence within a topic (Arun et al., 2010). It is also the one that maximises the word association between pairs of frequent words in each topic (Mimno et al., 2011) and maximises the coherence c_v measure, which calculates the similarity between every top word vector and the sum of all top word vectors (Röder et al., 2015). Therefore, since there was no point or range in the graph where all (or most) measures converged on their maximum or minimum point, the criterion to determine the number of topics was to find a point where all metrics had good values. On this basis, 17 topics were eventually selected.

The 17-topic LDA (see Appendix B, Table B2) was able to model ten topics that correlated with GT themes, where three of them were not in the 13-topic LDA, namely: a) reasons to join or leave a platform, b) miscommunication with platform management and c) teaching material and methods. However, there were still topics missing from the 17-topic LDA: a) how tutors consider the job, b) COVID-19-related discussions and c) discussions around new contracts.

In summary, the two LDA models were collectively able to detect twelve topics identified by GT analysis. However, the models failed to model one topic (COVID-19-related discussions) that was present in GT themes. Conversely, the only topic that was clearly modelled in both LDA models but not in GT was about the issue of bank transfers and transaction fees. Finally, since the aim of this step was to validate and further explore the data, and only one new topic was found even after the number of topics was increased to 17, the aim of this step was fulfilled, and it was not necessary to examine further models. Thus, the final output of this step is 14 topics that will next be considered in QDTM.

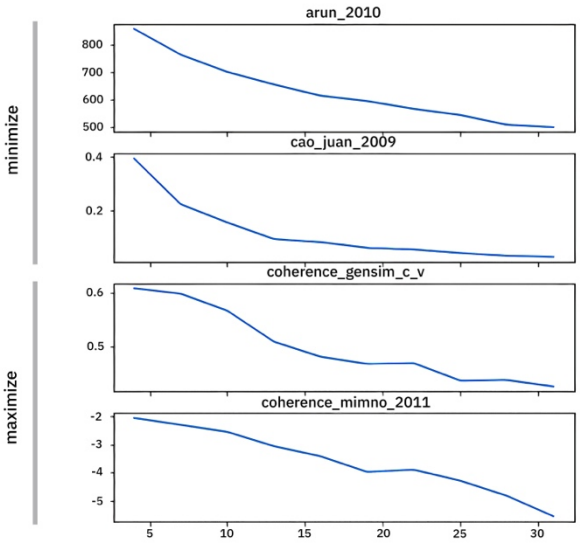

Figure 1. Evaluation of different LDA models when varying the number of topics K.

Topic Labelling – an explanatory example.

By way of illustration, this section describes the process of labelling LDA topics as explained in Phase One step (b). Taking topic1 (Appendix B, Table B2) as an example, in order to assign one or more labels to this topic, we began with a close reading of each of the five documents produced by LDA for this topic. It turned out that all the discussions in these documents (posts) revolved around one main topic. For example, in post 1, a tutor talked about their current schedule: "*I open 2 afternoon slots (Mon-Thurs) and 2 morning slots (Wed&Fri&Sat), so that's 13 in total.*". This individual also added information on when tutors can start to get a more steady schedule: "*once you convert the trials, the students will be on your schedule on a weekly basis, so you won't have to worry about bookings*". Similarly, in post 2, another tutor stated: "*I am doing 4-6 priority hours a day, and I have quite a few bookings this week*". Likewise, in post 3, a tutor shared their working hours and the way they schedule bookings: "*I work similar hours (15 hrs.) each week, so do my reservation schedule for a couple of weeks ahead*".

In the remaining two posts, the tutors mentioned days and times of the year with better or worse booking conditions. For instance, post 4 stated: "*Weekends always busy, but lately it's been*

quiet", and the tutor who published post 5 tried to explain why: "*It is the last week of school for many kids. I had a lot of cancellations from regulars. My worst day was Monday*".

It is apparent that all the discussions in this topic revolve around tutors' current booking status, their schedule and working hours, and bookings situation at different times. Thus, after understanding each post's content, we decided to label the topic as "Bookings and working hours" which we believe reflects the underlying discussion.

3.3 Term extraction results

The terms extracted for each topic that are needed for QDTM are presented in Appendix C, Table C1. As discussed earlier, the topic labels (shown in the first column) are derived from a close reading of topic documents. Then, the 20 most highly weighted terms for each topic (in both LDA models) were examined, and based on our understanding of the topic at hand, we selected the terms that we considered to be most relevant.

The first eight rows in Table C1 represent the topics that have been identified in both LDA models. Following the topic label column, the lists of terms are categorised according to whether they commonly appeared in both 13-topic and 17-topic LDA models, and subsequent columns show terms that appeared uniquely in either 13-topic or 17-topic LDA. Consider the topic "Hiring process" as an example, the terms "interview and apply" appeared in both LDA models, while the terms "referral, link, process and code" only appeared in 13-topic LDA, and the terms "email, profile and application" were uniquely appeared in 17-topic LDA.

The following five rows represent the topics found in only one of the LDA models as well as the terms extracted from them. While, in the last row, we propose terms for the topic "COVID-19-related discussions" since this topic does not appear in either of the LDA models.

4 Discussion and conclusions

In this section, we discuss the preliminary results from Phase One (GT and LDA analysis), which have allowed us to gain some early insights toward answering the research question. Therefore, these initial results should be interpreted with caution.

The data analysis showed that tutors in the gig economy seem to experience both platform-related and teaching-related problems. Platform issues include lack of bookings, poor pay, the opacity and the unfairness of rating systems, technical issues, and challenges in reaching the platform management. Comparatively, teaching-related issues appear to start from the first steps in joining these platforms, since platforms offering educational services tend to have a strict hiring process and job requirements. Other issues were found related to the unpaid time spent preparing lessons, as well as the time spent waiting for tutoring gigs to perform. Although tutors can depend on scheduled lessons to help them save time, it may limit their job autonomy and makes it more like a traditional work arrangement. Furthermore, the initial findings suggest that there are other challenges related to teaching that tutors may need to deal with, such as managing class time, dealing with students' different abilities and needs, and fulfilling student expectations. It seems that these teaching-related issues are distinctive of tutors' experiences in the gig economy, which shed light on some aspects of the second research question regarding how their experiences compare to those of other types of gig workers.

In addition to that, to perform tutoring gigs, tutors typically teach for quite long periods of uninterrupted focus, which might pose a barrier to entry for people with caregiving responsibilities or those who lack a quiet place to teach. This does not seem to conform to the assumption that location-independent platforms offer more freedom for workers to perform tasks around life activities. Nonetheless, the initial findings suggest that being location-independent has encouraged many people to join tutoring platforms during COVID-19 lockdowns, most likely due to an increase in free time or because of lost work or unemployment. Tutoring gigs, like most macrotasks, can help workers develop their skills and advance their careers. Furthermore, there is a possibility that tutoring platforms can help people without teaching experience explore the job's suitability. Tutoring online appears to be similar to working for location-dependent platforms in allowing workers to build interpersonal relationships and giving opportunities to develop longer-term relationships with students.

Finally, as only a small portion of the data has been analysed, our understanding of tutors' experience in the gig economy is still limited; we expect to gain more insights and a deeper understanding following our analysis of the data after the completion of phases 2 and 3 of the research plan.

References

Anwar, Mohammad Amir & Graham, Mark. 2021. Between a rock and a hard place: Freedom, flexibility, precarity and vulnerability in the gig economy in Africa. *Competition & Change,* 25(2)**,** 237-258.

Arun, Rajkumar, Suresh, Venkatasubramaniyan, Madhavan, CE Veni & Murthy, MN Narasimha. 2010. On finding the natural number of topics with latent dirichlet allocation: Some observations. In *Pacific-Asia conference on knowledge discovery and data mining*. Springer, pages 391-402.

Baumer, Eric PS, Mimno, David, Guha, Shion, Quan, Emily & Gay, Geri K. 2017. Comparing grounded theory and topic modeling: Extreme divergence or unlikely convergence? *Journal of the Association for Information Science and Technology,* 68(6)**,** 1397-1410.

Blei, David M, Ng, Andrew Y & Jordan, Michael I. 2003. Latent dirichlet allocation. *Journal of machine Learning research,* 3(Jan)**,** 993-1022.

Boussalis, Constantine & Coan, Travis G. 2016. Text-mining the signals of climate change doubt. *Global Environmental Change,* 36**,** 89-100.

Brookes, Gavin & McEnery, Tony. 2019. The utility of topic modelling for discourse studies: A critical evaluation. *Discourse Studies,* 21(1)**,** 3-21.

Broughton, Andrea, Gloster, Rosie, Marvell, Rosa, Green, Martha, Langley, Jamal & Martin, Alex. 2018. The experiences of individuals in the gig economy. *HM Government*.

Cano, Melissa Renau, Espelt, Ricard & Morell, Mayo Fuster. 2021. Flexibility and freedom for whom? Precarity, freedom and flexibility in on-demand food delivery. *Work Organisation, Labour & Globalisation,* 15(1)**,** 46-68.

Cao, Juan, Xia, Tian, Li, Jintao, Zhang, Yongdong & Tang, Sheng. 2009. A density-based method for adaptive LDA model selection. *Neurocomputing,* 72(7-9)**,** 1775-1781.

Charmaz, Kathy. 2006. *Constructing grounded theory: A practical guide through qualitative analysis.* sage.

Clark, Travis. 2021. The Gig Is Up: An Analysis of the Gig-Economy and an Outdated Worker Classification System in Need of Reform. *Seattle Journal for Social Justice,* 19(3)**,** 25.

DiMaggio, Paul. 2015. Adapting computational text analysis to social science (and vice versa). *Big Data & Society,* 2(2)**,** 2053951715602908.

Duggan, James, Sherman, Ultan, Carbery, Ronan & McDonnell, Anthony. 2021. Boundaryless careers and algorithmic constraints in the gig economy. *The International Journal of Human Resource Management***,** 1-31.

Dumais, Susan & Chen, Hao. 2000. Hierarchical classification of web content. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*. pages 256-263.

Edward, Webster. 2020. The Uberisation of work: the challenge of regulating platform capitalism. A commentary. *International Review of Applied Economics,* 34(4)**,** 512-521.

Fang, Zheng, He, Yulan & Procter, Rob. 2021. A Query-Driven Topic Model. In Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021. Online. Association for Computational Linguistics, pages 1764-1777. https://aclanthology.org/2021.findings-acl.154

Glavin, Paul, Schieman, Scott & Bridge, Council. 2021. Dependency and hardship in the gig economy: The mental health consequences of platform work. *Unpublished manuscript]. http://dx. doi. org/10.13140/RG,* 2(15287.65444).

Grimmer, Justin & Stewart, Brandon M. 2013. Text as data: The promise and pitfalls of automatic content analysis methods for political texts. *Political analysis,* 21(3)**,** 267-297.

Heeks, Richard, Graham, Mark, Mungai, Paul, Van Belle, Jean-Paul & Woodcock, Jamie. 2021. Systematic evaluation of gig work against decent work standards: The development and application of the Fairwork framework. *The Information Society***,** 1-20.

Howcroft, Debra & Bergvall-Kåreborn, Birgitta. 2019. A typology of crowdwork platforms. *Work, Employment and Society,* 33(1)**,** 21-38.

Jarrahi, Mohammad Hossein, Newlands, Gemma, Lee, Min Kyung, Wolf, Christine T, Kinder, Eliscia & Sutherland, Will. 2021. Algorithmic management in a work context. *Big Data & Society,* 8(2)**,** 20539517211020332.

Jarrahi, Mohammad Hossein & Sutherland, Will. 2019. Algorithmic Management and Algorithmic Competencies: Understanding and Appropriating Algorithms in Gig work. In *International Conference on Information*. Springer, pages 578-589.

Johnson, Stephen C. 1967. Hierarchical clustering schemes. *Psychometrika,* 32(3)**,** 241-254.

Kenney, Martin & Zysman, John. 2016. The rise of the platform economy. *Issues in science and technology,* 32(3)**,** 61.

Koutsimpogiorgos, Nikos, Van Slageren, Jaap, Herrmann, Andrea M & Frenken, Koen. 2020. Conceptualizing the Gig Economy and Its Regulatory Problems. *Policy & Internet,* 12(4)**,** 525-545.

Mimno, David, Wallach, Hanna, Talley, Edmund, Leenders, Miriam & McCallum, Andrew. 2011. Optimizing semantic coherence in topic models. In *Proceedings of the 2011 conference on empirical methods in natural language processing*. pages 262-272.

Morse, Janice M. 2015. Critical analysis of strategies for determining rigor in qualitative inquiry. *Qualitative health research,* 25(9)**,** 1212-1222.

Nelson, Laura K. 2017. Computational grounded theory: A methodological framework. *Sociological Methods & Research*, 0049124117729703.

Nemkova, Ekaterina, Demirel, Pelin & Baines, Linda. 2019. In search of meaningful work on digital freelancing platforms: the case of design professionals. *New Technology, Work and Employment,* 34(3), 226-243.

Rani, Uma & Furrer, Marianne. 2019. On-demand digital economy: Can experience ensure work and income security for microtask workers? *Jahrbücher für Nationalökonomie und Statistik,* 239(3), 565-597.

Röder, Michael, Both, Andreas & Hinneburg, Alexander. 2015. Exploring the space of topic coherence measures. In *Proceedings of the eighth ACM international conference on Web search and data mining*. pages 399-408.

Tan, Zhi Ming, Aggarwal, Nikita, Cowls, Josh, Morley, Jessica, Taddeo, Mariarosaria & Floridi, Luciano. 2021. The ethical debate about the gig economy: A review and critical analysis. *Technology in Society,* 65, 101594.

Woodcock, Jamie & Graham, Mark. 2019. The gig economy. *A critical introduction. Cambridge: Polity*.

Appendices

Appendix A. Grounded Theory Analysis – A brief description of the themes

1. **Hiring process:** Begins with an application form, then recording a demonstration class, passing a quiz and an interview. Tutors stated that the process is vague and changes frequently. Furthermore, applicants complained of not being informed of their progress, delays in results, and a lack of feedback after rejection.
2. **New contracts:** New tutors and tutors wanting to renew their contracts discussed contracts and related issues, such as increased emphasis on the rating system and pay cuts.
3. **Job requirements:** Being a native speaker is sometimes the only requirement. However, most platforms that provide educational services have stricter requirements, including holding a degree, teaching certificate and experience, and being legally eligible to work. Nonetheless, platforms' acceptance of tutors seems to depend mainly on their need for tutors at the time of applying.
4. **How tutors consider the job:** Tutors tend to view online tutoring as a part-time job to supplement their income, since it is impossible to make a living wage doing it.
5. **The class and the students:** Classes usually take the form of 25-minute one-to-one sessions. Students can be young children or adults. Tutors mostly seemed to have positive opinions of their students.
6. **Teaching material and methods:** When describing lessons, tutors discussed the material they taught and methods they used. Tutors are sometimes required to teach platform-provided materials.
7. **Bookings and working hours:** Bookings are arranged either by the platform or the student, who chooses a suitable tutor based on their profile and rating. Struggling to get bookings is a common problem; some suggest the solution is to be available at all times, which can be impractical and fatiguing. The instability in working hours, both daily and throughout the year, cancellations and student 'no-shows' are among the discussed problems.
8. **Payment:** The average payment is between $14 and $25/hr. Tutors can get bonuses for being on time, teaching during peak hours, or short-notice bookings. Related issues are inadequate payments and pay cuts.
9. **Rating system:** Tutors care deeply about their ratings, as they affect their payment and the number of bookings they receive. Another issue is the opacity of the ratings systems.
10. **Reasons to join or leave a platform:** Some important motives include making money, job autonomy, flexibility and suitability. Reasons to leave a platform include a lack of bookings, inadequate payment, and technical problems. Other reasons relate to the curriculum or feelings of boredom. Some tutors work on multiple platforms to overcome these problems.
11. **COVID-19-related discussions:** Tutors discussed the effects of the pandemic and how it encouraged them join these platforms, due to an increase in free time or because they lost their job. Furthermore, tutors reported that are receiving more bookings due to the closure of schools.
12. **Technical problems:** Not being able to log in, app crashing, or being double-booked are some examples. These can be more frustrating when occurring during lessons, as tutors must pause or cancel the class, which may negatively affect their ratings and payments.
13. **Miscommunication with platforms' management:** Long waiting times for the management team to response, or not receiving one at all, leading tutors to rely on Reddit for answers and solutions.
14. **Expressing feelings and sharing experiences:** Tutors tend to use Reddit forums to express their feelings and share their experiences, both positive and negative.
15. **Seeking and providing help and advice:** An important use of Reddit for tutors at different stages is to ask for or provide advice and help. Tutors seem to be honest, empathetic and generally supportive of each other.

Appendix B. Top-10 terms and topic labels for LDA models

1. 13-topic LDA

Topic.	Top-10 terms	Label*
Topic1	student, teacher, lesson, think, want, really, make, know, learn, people	The class and the students
Topic2	class, time, student, cancel, minute, book, take, get, day, week	Technical problems
Topic3	rating, month, student, get, year, new, week, go, class, contract	The new contracts
Topic4	teach, online, english, experience, school, tefl, course, degree, native, year	Job requirements
Topic5	kid, student, level, use, lesson, question, word, time, slide, class	The class and the students
Topic6	hour, time, week, day, work, schedule, book, class, open, slot	Bookings and working hours
Topic7	company, pay, teacher, work, hire, base, rate, esl, low,people	Payments
Topic8	parent, give, student, kid, say, f***, feedback, teacher, bad, star	Rating system/ The class and the students
Topic9	say, know, post, email, see, people, make, use, send, app	Technical problems
Topic10	pay, lesson, tutor, use, account, student, teacher, money, work, make	Bank transfers and transaction fees
Topic11	work, live, job, china, county, people, think, make, time, get	How tutors consider the job
Topic12	issue, problem, try, work, use, demo, internet, hope, test, good	Technical problems
Topic13	class, minute, min, referral, teach, pay, link, per, good, na	Hiring process

Table B1. Top-10 terms and topic labels for 13-topic LDA

*__Note:__ Topic labels are produced by reading the 5 highest weighted documents for each topic.

2. 17-topic LDA

Topic	Top-10 terms	Label*
Topic1	week, hour, day, time, book, class, slot, schedule, open, month	Bookings and working hours
Topic2	teach, online, work, hour, company, pay, class, experience, tefl, time	Payments/Job requirements
Topic3	student, teacher, work, class, give, rating, really, think, month, company	Rating system
Topic4	student, use, lesson, question, word, ask, say, make, learn, conversation	Teaching material and methods
Topic5	company, teacher, job, work, pay, people, make, money, online, think	Reasons to join or leave a platform
Topic6	class, minute, student, call, time, show, start, late, happen, reservation	Technical problems/ Bookings and working hours
Topic7	class, teacher, parent, time, student, contract, leave, cancel, take, kid	Bookings and working hours
Topic8	lesson, pay, time, student, hour, teacher, rate, tutor, base, minute	Payments
Topic9	kid, teach, level, well, old, year, think, really, feel, say	The class and the students
Topic10	know, anyone, video, tutor, help, ask, let, please, apply, interview	Hiring process / Miscommunication with platform management
Topic11	email, send, message, say, get, group, reply, back, try, see	Hiring process
Topic12	good, student, make, bad, star, sound, say, give, wow, know	The class and the students
Topic13	people, post, think, say, know, f***, mean, name, much, comment	Random
Topic14	english, native, live, country, speaker, language, china, work, american, non	Job requirements
Topic15	month, pay, use, app, work, get, th, phone, tax, year	Payments/ Technical problems
Topic16	feedback, parent, review, keep, student, want, know, class, courseware, see	Rating system
Topic17	rating, account, pay, demo, use, bank, test, paypal, payment, internet	Bank transfers and transaction fees

Table B2. Top-10 terms and topic labels for 17-topic LDA

*__Note:__ Topic labels are produced by reading the 5 highest weighted documents for each topic.

Appendix C. Terms per topic required to apply QDTM

	Topic label	Common terms in both 13- and 17-topic LDA	Terms unique to 13-topic LDA	Terms unique to 17-topic LDA	
1	**Hiring process**	interview, apply	referral, link, process, code	email, profile, application	**Topics appeared in both 13- and 17-topic LDA.**
2	**Job requirements**	experience, native, degree, tefl, esl, course, company	certificate	country, speaker, language, live, hire, require	
3	**The class and the students**	kid, student, level, lesson, class, time, call, teach	video, slide, read, conversation	child, late, show, start, camera, wait, young	
4	**Bookings and working hours**	schedule, class, book, slot, hour, time, week, day, month, open, weekend, booking	-	leave, cancel, bonus, trial, regular, ph, cancelation	
5	**Payments**	rate, base, pay, low, make	hire, high, offer	price, tax, per	
6	**Rating system**	rating, give, feedback, review, bad	star	parent, comment, assessment, good	
7	**Technical problems**	app, computer	issue, problem, try, test, connection, internet, email, send, post, check	camera	
8	**Bank transfers and transaction fees**	bank, account, pay, paypal, payment	money, platform, price, charge	transfer, payoneer, fee	**The new topic, absent from GT**
9	**The new contracts**	N/A	contract, rating, new, change, year, start	N/A	**Topics appeared only in either 13- or 17-topic LDA**
10	**How tutors consider this job**	N/A	work, live, job, time, money, need, life, income	N/A	
11	**Teaching material and methods**	N/A	N/A	use, question, conversation, learn, ask, slide, talk, answer, level, write, read	
12	**Reasons to join or leave a platform**	N/A	N/A	job, work, pay, make, money, online, business	
13	**Miscommunication with platform management**	N/A	N/A	contact, ticket, response, email, send	
		Proposed Terms			
14	**COVID-19-related discussions**	pandemic, COVID-19, lockdown			**The missing topic from both LDA models**

Table C1. Terms classified as either common terms generated by both 13- and 17-topic LDA, terms that only appeared in one model or the other, and proposed terms for the missing topic from LDA.

Can Domain Pre-training Help Interdisciplinary Researchers from Data Annotation Poverty? A Case Study of Legal Argument Mining with BERT-based Transformers

Gechuan Zhang[1], **David Lillis**[1] **and Paul Nulty**[2]
[1] School of Computer Science, University College Dublin, Ireland
[2] Department of Computer Science, Birkbeck, University of London, UK
gechuan.zhang@ucdconnect.ie
david.lillis@ucd.ie
p.nulty@bbk.ac.uk

Abstract

Interdisciplinary Natural Language Processing (NLP) research traditionally suffers from the requirement for costly data annotation. However, transformer frameworks with pre-training have shown their ability on many downstream tasks including digital humanities tasks with limited small datasets. Considering the fact that many digital humanities fields (e.g. law) feature an abundance of non-annotated textual resources, and the recent achievements led by transformer models, we pay special attention to whether domain pre-training will enhance transformer's performance on interdisciplinary tasks and how. In this work, we use legal argument mining as our case study. This aims to automatically identify text segments with particular linguistic structures (i.e., arguments) from legal documents and to predict the reasoning relations between marked arguments. Our work includes a broad survey of a wide range of BERT variants with different pre-training strategies. Our case study focuses on: the comparison of general pre-training and domain pre-training; the generalisability of different domain pre-trained transformers; and the potential of merging general pre-training with domain pre-training. We also achieve better results than the current transformer baseline in legal argument mining.

1 Introduction

Interdisciplinary natural language processing (NLP) has become one of the most important trends in NLP development. For example, processing of legal text has resulted in research topics such as legal topic classification (Nallapati and Manning, 2008), legal information extraction (Chalkidis et al., 2018), judicial decision prediction (Chalkidis et al., 2019), and legal argumentation mining (Mochales and Moens, 2011). Among these, legal argumentation mining is especially difficult, but has strong application potential, given that arguments are among the most important language structures used in the law. The goal of argument mining is to automatically detect arguments from raw text as well as to identify the reasoning relationships between these arguments (Mochales and Moens, 2011). Argument mining systems that help to identify and analyse argumentative text can assist legal professionals to save time and effort when facing modern document systems with huge quantities of files.

However, the shortage of suitable datasets and the high cost of annotating new datasets impede the application of many advanced NLP approaches (such as neural network and deep learning) to legal argument mining, which is a common issue in most interdisciplinary NLP research. Creating and constructing annotated corpora is complex and labour intensive (Lippi and Torroni, 2016; Poudyal et al., 2020). Particularly when the raw text is domain-specific (e.g., legal text), the annotating experts are required to have extensive knowledge of the corresponding field. This leads to a paradoxical situation whereby a domain with enormous quantities of text resources built up over centuries is served by only a small number of suitable corpora that tend to be limited in their scale.

This dilemma in interdisciplinary research may be solved by using transformer frameworks, such as BERT (Devlin et al., 2019): first pre-training (self-supervised learning) on a large group of roughly labelled text, then fine-tuning on downstream tasks with much smaller datasets that do fine-grained feature annotation. Transformers have revolutionised many research fields including legal text processing (Chalkidis et al., 2019; Reimers et al., 2019). This caused us to examine the potential of reducing the burden of annotation in interdisciplinary research through pre-training. In this work, we use legal argument mining as our case study, because it includes not only the general text classification, but also the relation mining on legal text, and has a

Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH), pages 121–130
December 19, 2021.

strong connection with both the legal field (humanities) and the study of argument mining (NLP).

The primary aim of this paper is to explore the extent to which domain pre-training (i.e. pre-training transformers using legal texts) can improve transformers' performance on legal text processing tasks, without the need for large volumes of expensive annotation. Legal text has its own distinct characteristics when compared with general English-language writing, this also motivates an investigation as to whether legal-specific pre-training can improve upon transformers pre-trained on general-purpose corpora (e.g. Wikipedia, books).

Although Poudyal et al. (2020) set a legal argument mining baseline using RoBERTa (Liu et al., 2019), none of the legal-domain pre-trained transformers (Chalkidis et al., 2020; Zheng et al., 2021) have been applied to legal argument mining tasks to date. In this case study, we try to find (a) whether domain pre-trained transformers outperform general pre-trained transformers in interdisciplinary NLP tasks; (b) whether domain pre-trained transformers maintain good generalisability when applied to tasks using another domain-specific dataset without overlap with the pre-train corpora; and (c) whether merging pre-train corpora from different domains can enhance the transformer performance on interdisciplinary downstream tasks.

In our work, we first provide a thorough survey including a wide range of BERT variants which emphasise two pre-train domains (generic and legal) and use different pre-train strategies. Then we evaluate these transformer models on three legal argument mining tasks: (1) argument clause recognition, (2) argument relation mining, (3) argument component recognition. We discuss the potential of using domain-specific pre-training to adapt state-of-the-art transformer models to interdisciplinary research that lacks large annotated datasets, and we analyse what adjustments in domain-specific pre-training may improve transformers' performance in a complex text processing problem like legal argument mining.

2 Argument and Argument Mining

Long before being treated as a research area in NLP, philosophers and rhetoricians had paid special attention to the logic and reasoning processes embodied in human languages. Numerous schemes and theories have been proposed to define and reason about argumentation. In our work, we use the definition given by Walton (2009) that an *argument* is a set of statements (propositions) that includes three parts: conclusion, premises and inference.

As described in (Lawrence and Reed, 2020), *argument mining* is the automatic identification and extraction of the structure of inference and reasoning expressed as arguments presented in natural language. There are two crucial stages in the framework of argument mining: *argument extraction* and *relation prediction* (Cabrio and Villata, 2018). Argument extraction is the first stage where the arguments (with their internal structures) are identified from the input documents. Relation prediction is where the support or attack relations between the arguments are predicted.

2.1 Structured Argumentation Model

Structured argumentation is one of the main approaches in computational argumentation, which presents an internal structure for each argument, described in terms of some knowledge representation. For structured argumentation models typically applied in argument mining tasks, defining the internal structure of an argument is crucial (Lippi and Torroni, 2015). Such models consider different argument components inside each argument and both internal and external argument relationships:

- **Argument Component** is the smallest unit in structured argumentation model. The argument components connect to each other through the internal relations.

- **Argument Relation** has two different levels in a structured argumentation model: internal and external. Internal argument relations are used to connect elementary argument components into a whole group (i.e., each argument). External argument relations represent the reasoning process between different arguments in a complete text document.

2.2 Walton Argumentation Model

The typical guideline for annotating legal argument mining corpora is Walton's structured argumentation model (Walton, 2009). This has two types of argument components: premises and conclusions. Walton (2009) described each argument as a set of statements (propositions) made up of three parts: a *conclusion*, a set of *premises*, and an *inference* from the set of premises to the conclusion. The conclusion is a claim or a statement which acts as the central component of an argument. The set

of premises are the evidences or reasons given to support the conclusion. The inference is the internal argument relation in the Walton argumentation model. For the external argument relations, Walton (2009) defined a set of bipolar relations: an argument can be supported by other arguments, or it can be attacked by other arguments (through raising critical questions about it).

3 BERT-Based Transformers

BERT (Bidirectional Encoder Representations from Transformers) is a contextual word embedding model using the deep transformer architecture (Vaswani et al., 2017) to derive word features (Devlin et al., 2019). It leverages a two-step framework: pre-training and fine-tuning. During pre-training, BERT is trained on a large corpus using self-supervised learning methods. Then, in fine-tuning, the model is tuned on the downstream task's dataset, which is usually much smaller. BERT has achieved state-of-the-art performance on many legal text processing tasks (Chalkidis et al., 2019; Reimers et al., 2019; Poudyal et al., 2020). Since then, several studies have addressed whether pre-training on legal texts can improve transformers' performance on downstream tasks in the same domain (Elwany et al., 2019; Chalkidis et al., 2020; Zhong et al., 2020a,b; Zheng et al., 2021), but none have been evaluated on legal argument mining.

4 BERT Pre-Train Strategies

In order to analyse how different pre-trained transformers perform on legal argument mining tasks, we compare five BERT$_{base}$ (L=12, H=768, A=12, 110M params (Devlin et al., 2019)) variants from two pre-train categories: general pre-trained models using generic English corpora for pre-training, and domain pre-trained models using English legal text in their pre-train corpus. Here we first provide an outline background of each transformer. Then, based on previous studies (Devlin et al., 2019; Liu et al., 2019), we compare these models across three key aspects of pre-train strategy: pre-train corpora selection, pre-train procedure, and text encoding.

4.1 BERT-based Transformers

RoBERTa
A widely used BERT$_{base}$ variant, pre-trained on large generic English corpora (Liu et al., 2019), which constitutes the latest baseline model for legal argument mining (Poudyal et al., 2020).

LEGAL-BERT Family
Legal-BERT$_{echr}$ and Legal-BERT$_{base}$ are two domain pre-trained transformers selected from the LEGAL-BERT family (Chalkidis et al., 2020). Instead of using general English corpora, the LEGAL-BERT family, a group of legal-specific BERT$_{base}$ variants, are pre-trained on a English legal text collection (see Table 1) with two different domain-adaptation methods: (a) further pre-train BERT$_{base}$ on legal text before fine-tuning, (b) pre-train BERT$_{base}$ from scratch on legal text before fine-tuning. More precisely, Legal-BERT$_{echr}$ is further pre-trained on the European Court of Human Rights (ECHR) legal case subset (see Table 2) from the BERT$_{base}$ checkpoint. Legal-BERT$_{base}$ is pre-trained from scratch on the whole collection of legal corpora.

Harvard Legal-BERT Variants
Distinct from the LEGAL-BERT family, two other domain-specific BERT$_{base}$ transformers from (Zheng et al., 2021) are pre-trained with the Harvard Law case corpus (see Table 1). To avoid name confusion, we use *Legal-BERT*$_{harv}$ and *Custom Legal-BERT*$_{harv}$ to represent Legal BERT and Custom Legal BERT in the original literature. Similar to Chalkidis et al. (2020), Zheng et al. (2021) also assess both *further pre-training* and *pre-training from scratch* domain-adaptation methods. Similar to Legal-BERT$_{echr}$, Legal-BERT$_{harv}$ is further pre-trained on the Harvard Law case corpus. Custom Legal-BERT$_{harv}$ is pre-trained from scratch using the same corpus with a custom vocabulary (see Section 4.4).

4.2 Pre-train Corpora Selection

In order to extract long contiguous sequences, both datasets in the BERT$_{base}$ original pre-train corpora are long documents (i.e., books and Wikipedia passages). Since Liu et al. (2019) suggest that BERT$_{base}$ is still under-trained, RoBERTa enlarged the scale of its pre-train corpora 10 fold (from 16 GB to 161 GB) by including news articles and online discussion web text. As for the domain pre-train corpora, Chalkidis et al. (2020) collected a wide range of English legal documents and cases with different functions, backgrounds, and text formats (i.e., legislation, case judgements, legal contracts). Zheng et al. (2021) focused on US legal decisions from the Harvard Law case corpus and gather a larger dataset (37 GB), which is three times

larger than the LEGAL-BERT family (11.5 GB).

Model	Text Type	Size (GB)
RoBERTa	BooksCorpus Wikipedia	16
	CC-News	76
	OpenWebText	38
	STORIES	31
	total	**161**
Legal-BERT$_{base}$	EU legislation	1.9
	UK legislation	1.4
	ECJ case	0.6
	ECHR case	0.5
	US court case	3.2
	US contract	3.9
	total	**11.5**
Custom Legal-BERT$_{harv}$	Harvard Law case	**37**

Table 1: Different BERT Variant Pre-train Corpora

Model	P Corpus	FP Corpus
Legal-BERT$_{echr}$	BooksCorpus Wikipedia 16 GB	ECHR case 0.5 GB
Legal-BERT$_{harv}$		Harvard Law case 37 GB

Table 2: Different BERT Variant Pre-train (P) Corpora and Further Pre-train (FP) Corpora

Because Legal-BERT$_{echr}$ and Legal-BERT$_{harv}$ are first initialised from the BERT$_{base}$ checkpoint, which has been pre-trained on generic corpora, then further pre-trained on legal corpora, both variants have mixed pre-train corpora. Legal-BERT$_{harv}$ uses the same legal corpora as Custom Legal-BERT$_{harv}$ for further pre-training. The ECHR case sub-set used by Legal-BERT$_{echr}$ for further pre-training is only 0.5 GB, which is much smaller compared to other pre-train corpora.

4.3 Pre-train Procedure

The original BERT$_{base}$ pre-train procedure contains two objectives: masked language modelling (MLM), which aims to train the model for a deep bidirectional representation, and next sentence prediction (NSP) which aims to train the model for sentence relationship understanding (Devlin et al., 2019). In place of performing MLM once during data pre-processing (static masking), RoBERTa generates the masking pattern for each sequence input (dynamic masking) and removes the NSP loss (Liu et al., 2019). The LEGAL-BERT family use the same pre-train objectives as the original BERT$_{base}$ (Chalkidis et al., 2020). The Harvard Legal-BERT variants make adjustments based on the characteristics of legal corpora. In constrast with BERT$_{base}$, which selects and replaces the tokens in the input sequence, Zheng et al. (2021) use whole word masking in MLM and adds regular expressions to ensure legal citations are included as part of a segmented sentence in NSP.

Liu et al. (2019) suggest that training the BERT model longer with larger batches improves its performance. The original pre-training setup of BERT$_{base}$ is 1M steps and 256 sequences per batch. RoBERTa replaces this with 500K steps and a batch size of 8K. For the two domain pre-trained models from the LEGAL-BERT family, Legal-BERT$_{base}$ uses the same setup as BERT$_{base}$ when being pre-trained from scratch; while Legal-BERT$_{echr}$ is first initialised from BERT$_{base}$'s 1M checkpoint then further pre-trained with another 5K on legal text. Like RoBERTa, Zheng et al. (2021) train the model longer by using 2M total pre-train steps for both Harvard Legal-BERT variants. In particular, Custom Legal-BERT$_{harv}$ is pre-trained from scratch with 2M steps, and Legal-BERT$_{harv}$ is initialised from the 1M checkpoint of BERT$_{base}$ (same as Legal-BERT$_{echr}$) then further pre-trained with 1M steps on legal case documents (see Table 2 and 3).

4.4 Text Encoding

To encode text pieces into vectors, the BERT$_{base}$ transformer first splits the input raw text into words or sub-words through a tokenizer. These word pieces are then converted to ids by using pre-designed vocabularies. The original BERT$_{base}$ is implemented with the WordPiece tokenizer (Schuster and Nakajima, 2012) and a character-level Byte-Pair Encoding (BPE) (Sennrich et al., 2016) vocabulary (size 30K). Both Legal-BERT$_{echr}$ and Legal-BERT$_{harv}$ use the same tokenizer and BPE vocabulary from BERT$_{base}$ during the further pre-training. Rather than using character-level sub-word unit, RoBERTa's implementation uses the same tokenizer as (Radford et al., 2018) with a larger byte-level BPE vocabulary (size 50K). Using byte-level BPE makes it possible to encode any input text without introducing "unknown" tokens.

In order to adapt BERT$_{base}$ from generic English corpora to legal text, both Legal-BERT$_{base}$ and Custom Legal-BERT$_{harv}$, apply the SentencePiece (Kudo and Richardson, 2018) tokenizer with self-generated vocabularies. Legal-BERT$_{base}$ used a newly-created vocabulary of equal size to BERT$_{base}$ (30K), constructed on its complete pre-train legal corpus (see Table 1). Custom Legal-

Model	Training Objectives	Pre-train Setup			Further Pre-train Setup			Encoding
		Type	Step	Batch	Type	Step	Batch	
RoBERTa	Dynamic MLM	Generic	500K	8K	-	-	-	Byte BPE
Legal-BERT$_{base}$	MLM, NSP	Legal	1M	256	-	-	-	SP
Legal-BERT$_{echr}$		Generic	1M	256	Legal	5K	256	WP
Custom Legal-BERT$_{harv}$	Whole-Word MLM,	Legal	2M	256	-	-	-	SP
Legal-BERT$_{harv}$	Regexp NSP	Generic	1M	256	Legal	1M	256	WP

Table 3: Different BERT Variant Pre-train Design (SP = SentencePiece, WP = WordPiece)

BERT$_{harv}$ also uses a legal domain-specific vocabulary (32K), which is constructed on a sub-sample of sentences of the Harvard Law case corpus.

5 ECHR Dataset

The European Court of Human Rights (ECHR) case-law dataset, developed from the HUDOC database[1], is an open-source database of case documents, and has become one of the most commonly-used datasets for legal text processing research such as judicial decision prediction (Chalkidis et al., 2019; Medvedeva et al., 2020), court decision event extraction (Filtz et al., 2020), and legal argument mining (Mochales and Moens, 2011; Teruel et al., 2018; Poudyal et al., 2020).

The ECHR case-law dataset has been used for legal argument mining research from an early stage (Mochales and Moens, 2011). (Mochales and Moens, 2008) provides a detailed structural analysis of ECHR documents. Several legal argument mining corpora have been established based on the ECHR case-laws (Mochales and Moens, 2011; Teruel et al., 2018; Poudyal et al., 2020). Among them, we choose the recently released ECHR argument mining corpus (ECHR-AM) (Poudyal et al., 2020) for our experiments. The ECHR-AM corpus contains 42 cases, 20 decisions (the average word length is 3,500 words) and 22 judgements (the average word length is 10,000 words). The entire corpus is annotated at the sentence level using three labels according to the Walton Argumentation Model (see Section 2.2): premise, conclusion, and non-argument. The annotation focuses on internal argument relations which includes a total of 1,951 premises and 743 conclusions acting as argument components for individual arguments.

6 Legal Argument Mining Tasks

Our case study currently focuses on the argument extraction, which is the first stage within a typical argument mining framework (as mentioned in Section 2). Following the example of (Poudyal et al., 2020), we organise this as three tasks.

[1] http://hudoc.echr.coe.int/

[Non-Argument] *Article 5 paras. 3 and 4 (Art. 5-3, 5-4) provide certain guarantees of judicial control of provisional release or detention on remand pending trial.*
[Premise] *The Commission notes that the applicant was detained after having been sentenced by the first instance court to 18 months' imprisonment.*
[Premise] *He was released after the Court of Appeal reviewed this sentence, reducing it to 15 months' imprisonment, convertible to a fine.*
[Conclusion] *The Commission finds that the applicant was deprived of his liberty "after conviction by a competent court" within the meaning of Article 5 para. 1 (a) (Art. 5-1-a) of the Convention.*

Figure 1: Annotation Example of the ECHR Argument Mining Corpus

6.1 Argument Clause Recognition

The first task is to filter those sentences that are argumentative from those that are not. We treat this task as a binary text classification, in which the segmented clauses from the case law documents are classified into two groups: argument clauses and non-argument clauses. The argument clauses are those sentences which functionally act as argument components in arguments (see Section 2.2).

6.2 Argument Relation Mining

This task focuses on identifying the argument relations that link argument components (i.e., argument clauses) within each argument. Here, the *argument relation* is the internal relation (i.e., inference) in the Walton argumentation model. The ultimate goal of this task is to label argument clauses that appear in the same argument as being in the same group. Since the same argument clause may appear in different arguments (for example, a single clause can be the conclusion of one argument and also the premise of another), this task is more difficult than a general text clustering problem. Previous studies imply this task is probably the bottleneck in the argument mining framework (Mochales and Moens, 2011; Poudyal et al., 2019, 2020).

Instead of directly grouping argument clauses into individual arguments, we consider the solution in (Poudyal et al., 2020) and treat this task as an argument clause pair classification task. We analyse whether or not a pair of clauses are argument components from the same argument. This can help with the multi-correspondence issue between argument clauses and arguments. To get the input argument clause pairs, we order the argument clauses from the same case document into a sequence, then use a sliding window to generate input clause pairs. Next, we use transformers to predict whether the pair are related.

6.3 Argument Component Classification

The final task is to classify the argument clauses as premises or conclusions. Because an argument clause may belong to multiple arguments, and can be either a premise or a conclusion, we separated the argument component classification task into two individual binary classification sub-tasks (premise recognition and conclusion recognition). More specifically, if an argument clause is tagged as both premise and conclusion in the classification sub-tasks, it is included in multiple arguments. As a conclusion, the argument clause itself represents an individual argument connecting with other premise clauses. As a premise, this argument clause is also a part of an argument, whose conclusion is linked with this clause in the argument relation mining task.

7 Experiments

7.1 Experimental Setup

Baseline: Following the baseline setup given by (Poudyal et al., 2020), we use 5-fold cross validation during our experiment. We split 80% case law documents for training and the remaining 20% for testing. Of the training documents, we randomly select 20% for validation. The number of documents in each train-validation-test split is therefore 28-6-8. During each fold, we select the model with the best F-score on the validation set for testing. We performed five runs for each model and reported mean evaluation scores with standard deviations. For the baseline, we refer to the records in Poudyal et al. (2020) and use RoBERTa for extra tests (in argument clause recognition and argument relation mining). Moreover, to better understand the enhancement given by the BERT model, we also include an additional non-BERT baseline. A number of candidate approaches based on word embeddings (Wang and Lillis, 2020) were considered. We choose the one-layer BiLSTM architecture used by Zheng et al. (2021), tested with 300-dimension word embedding. For each task, we first encode the segmented clause with the transformer, then pool the final CLS token from the embedding vector and input it into the classifier head. For each selected BERT-based transformer in the experiment, we add the same classifier head containing a dropout layer (dropout rate = 0.1) and a liner layer (for the final classification task).

Hyper-parameters: Because Poudyal et al. (2020) do not provide full details of the hyper-parameters used in their experiments, we consult the hyper-parameter setups in (Devlin et al., 2019; Chalkidis et al., 2020; Zheng et al., 2021) for guidance on typical experiment configurations. Similar to Zheng et al. (2021), we perform the first round of grid search for learning rate in the range {2e-5, 3e-5, 4e-5, 5e-5} suggested by Devlin et al. (2019), then we expand this range with {5e-6, 1e-5, 6e-5} to test the boundary. Considering the small size of the ECHR-AM corpus, we search over batch size {8, 16, 32} as recommended by Chalkidis et al. (2020). Poudyal et al. (2020) set 15 fine-tune epochs for each baseline task, which we found redundant due to the fact that the BERT-base transformers are well trained and can adapt quickly on small corpora. We fine-tune each model with 4 epochs in each task.

7.2 Argument Clause Recognition Results

As is mentioned in Poudyal et al. (2020), over 99% of the argument clauses of the ECHR-AM dataset are present in a specific section ("AS TO THE LAW/ THE LAW") of each document. The baseline argument clause recognition task first uses text matching to detect target sections, then classifies segmented sentences (clauses outside that section are automatically predicted as non-argument). The upper part of Table 4 shows the results for the argument clause recognition after section detection. The baseline (from Poudyal et al. (2020); generated by RoBERTa) was outperformed by all the domain pre-trained transformers. Both models from the LEGAL-BERT family reached the highest F1 score (79.3%), which is probably because their pre-train text collection includes ECHR cases. Among

all the domain pre-trained transformers, it is impressive that Legal-BERT$_{echr}$ only used 0.5 GB legal text (12K ECHR cases) in its further pre-train and gained a competitive performance on the argument clause recognition task. Further pre-trained Legal-BERT$_{harv}$ also reached a better precision (74.3% vs. 69.7%) than the RoBERTa baseline. Although BiLSTM achieves the highest recall, its relatively low precision leads to an F1 score that is below the BERT-based models.

Legal Sect	**P (%)**	**R (%)**	**F1 (%)**
baseline	69.7	84.8	76.5
BiLSTM	62.4±6.5	**91.8**±4.4	74.0±3.5
Legal-BERT$_{base}$	72.4±2.4	88.1±1.4	**79.3**±1.3
Legal-BERT$_{echr}$	73.5±2.1	86.5±2.2	**79.3**±1.3
C-Legal-BERT$_{harv}$	73.4±1.9	84.2±1.9	78.2±1.8
Legal-BERT$_{harv}$	**74.3**±1.8	84.0±0.8	78.7±1.0
Whole Doc	**P (%)**	**R (%)**	**F1 (%)**
RoBERTa	65.3±1.8	71.0±4.5	67.7±2.4
BiLSTM	61.0±7.1	51.4±7.5	55.5±6.1
Legal-BERT$_{base}$	66.1±1.8	**73.6**±3.9	69.3±2.2
Legal-BERT$_{echr}$	**67.5**±2.0	73.1±2.7	**69.9**±2.1
C-Legal-BERT$_{harv}$	65.3±2.8	69.8±3.9	67.1±2.3
Legal-BERT$_{harv}$	66.3±1.7	70.0±3.2	67.9±1.9

Table 4: Precision (P), recall (R), F1 measurement (± std. dev.) for the argument clause recognition task on the "AS TO THE LAW/ THE LAW" Section scope and the whole document scope (C = Custom).

The section detection in the baseline argument clause recognition task filters out a large group of non-argument clauses, and balances the candidate clause dataset. The number of input segmented clauses shrank from 10,456 to 4,683. To generalise this approach to practical applications, we expand the searching area to the complete document (10,456 clauses) and test again. The results are displayed in the lower part of Table 4. All domain pre-trained models exceeded the RoBERTa baseline, except the Custom Legal-BERT$_{harv}$ whose score is slightly lower (67.1% vs. 67.7%). Legal-BERT$_{echr}$ remained the best F1 score (69.9%). Among the four legal-specific transformers, the models pre-trained with further steps had slightly better scores than the other two models pre-trained from scratch. Considering the scale of each model's pre-train corpus: RoBERTa was pre-trained on 161 GB text, while other BERT models used much smaller pre-train corpora. The evaluation scores in this task suggest that domain-specific pre-training is effective when downstream tasks in argument mining (e.g., text classification) are focusing on text with a similar domain-specific background.

7.3 Argument Relation Mining Results

To be consistent with Poudyal et al. (2020), we assume that all the argument clauses have been successfully identified from previous task. As discussed in Section 6.2, we use a sliding window on the argument clause sequence to generate pairs of argument clauses. In order to compare the RoBERTa baseline and different domain pre-train variants, we first use the window size 5 mentioned in Poudyal et al. (2020). The upper part of Table 5 shows the results with domain pre-training again displaying its effectiveness when mining relations between clause pairs. All the domain pre-trained transformers substantially exceeded the baseline F1 score (8.4% on average). The Harvard Legal BERT variants slightly outperformed the LEGAL-BERT family in each corresponding pre-train type (pre-train from scratch, 59.9% vs. 58.1%; further pre-train, 60.6% vs. 59.4%). Among the four domain-specific pre-trained transformers, models using further pre-train strategy again displayed a further slight advantage. By using the further pre-train approach and adding special pre-training adjustments (whole word MLM, Regexp NSP, see Section 4.3), Legal-BERT$_{harv}$ reached the best scores for precision (59.7%), recall (62.1%) and F1 (60.6%).

window size = 5	**P (%)**	**R (%)**	**F1 (%)**
baseline	50.2	52.1	51.1
BiLSTM	45.5±5.5	52.7±14.6	47.8±6.1
Legal-BERT$_{base}$	57.3±2.4	59.7±3.5	58.1±2.6
Legal-BERT$_{echr}$	59.5±3.8	60.0±3.8	59.4±1.2
C-Legal-BERT$_{harv}$	58.7±2.4	61.2±1.6	59.9±1.8
Legal-BERT$_{harv}$	**59.7**±2.6	**62.1**±2.5	**60.6**±1.6
window size = 10	**P (%)**	**R (%)**	**F1 (%)**
RoBERTa	**47.2**±3.6	35.2±1.6	39.4±2.5
BiLSTM	26.3±3.8	**45.8**±20.7	31.3±7.3
Legal-BERT$_{base}$	45.8±4.9	36.4±4.3	39.6±1.8
Legal-BERT$_{echr}$	46.8±3.2	41.4±1.8	43.3±1.3
C-Legal-BERT$_{harv}$	45.8±2.3	41.8±2.4	43.2±2.1
Legal-BERT$_{harv}$	47.1±2.4	43.6±2.9	**44.5**±2.0

Table 5: Precision (P), recall (R), F1 measurement (± std. dev.) for the argument relation mining task with window size = 5 and window size = 10 (C = Custom).

After analysing the ECHR argument mining dataset, we decided to enlarge the window size to 10, in which almost all the actual argument relations are included, while the number of total pairs has not increased to an unmanageable degree (by doubling the window size, the total number of argument clause pairs increased from 10,356 to 22,329). Due to the imbalance of the argument clause pair dataset, the general performance of all transformers were lower as expected. Legal-BERT$_{harv}$ again

reached the best F1 score (44.5%).

7.4 Argument Component Classification Results

As mentioned in Section 6.3, the argument component classification task consists of two sub-tasks: premise recognition and conclusion recognition. In the same way as (Poudyal et al., 2020), we assume that we have successfully identified argument clauses in the previous task. For the premise recognition sub-task, domain pre-training improves the BERT-base model's performance over all three evaluation scores, as shown in Table 6: Custom Legal-BERT$_{harv}$ reaches both highest recall (91.5% vs. 88.7%) and F1 (87.2% vs. 85.9%) among all the transformers; Legal-BERT$_{harv}$ also achieves a moderate improvement in precision (83.9% vs. 83.2%). Compared to the BERT-based models, BiLSTM has the highest recall value but a much lower precision, and is less robust across runs.

Premise	**P (%)**	**R (%)**	**F1 (%)**
baseline	83.2	88.7	85.9
BiLSTM	79.2±2.7	**94.6**±2.3	86.2±1.4
Legal-BERT$_{base}$	83.8±1.4	90.3±1.8	86.8±0.7
Legal-BERT$_{echr}$	83.5±2.5	90.4±1.7	86.7±0.9
C-Legal-BERT$_{harv}$	83.4±1.3	91.5±1.3	**87.2**±0.6
Legal-BERT$_{harv}$	**83.9**±1.6	90.3±1.8	86.9±0.9
Conclusion	**P (%)**	**R (%)**	**F1 (%)**
baseline	58.9	**67.2**	62.8
BiLSTM	54.9±11.3	54.0±11.2	52.6±4.7
Legal-BERT$_{base}$	65.2±2.8	61.2±5.2	61.9±3.0
Legal-BERT$_{echr}$	65.3±1.1	62.0±3.5	63.3±2.3
C-Legal-BERT$_{harv}$	**67.1**±0.9	60.1±3.7	62.9±2.0
Legal-BERT$_{harv}$	66.2±0.9	63.1±3.5	**64.2**±1.8

Table 6: Precision (P), recall (R), F1 measurement (± std. dev.) for the argument component (premise/conclusion) classification task (C = Custom).

For the conclusion recognition sub-task, Legal-BERT$_{harv}$ outperforms the RoBERTa baseline (64.2% vs. 62.8%) with the highest recall (63.1%) among all domain pre-trained transformers. Custom Legal-BERT$_{harv}$ also reaches the best precision (67.1%). Generally, pre-trained legal-BERT models show better precision than recall, in contrast to the baseline (58.9% precision, and 67.2% recall). Since (Poudyal et al., 2020) does not provide multiple cross-validation records, we suggest that this difference may be a result of randomness bias. Overall, the Harvard Legal BERT variants slightly outperformed the LEGAL-BERT family.

8 Discussion

Our case study on legal argument mining gives us insights on the potential of using domain pre-training to reduce the data annotation burden in interdisciplinary NLP research, as well as help us better understand the relationship between domain pre-training and domain-specific downstream tasks. To answer question (a) in Section 1, it is clear that domain pre-trained transformers work better than general pre-trained transformers in all three legal argument mining tasks, which also exceed the baseline from (Poudyal et al., 2020). This supports the idea that domain pre-training helps improve transformer's performance on downstream tasks where only small datasets are available. Combining the scope of pre-train corpora used by each transformer with its performance, we suggest that using small domain-specific pre-training corpora would be as effective as using a large general corpora. In Section 7, the LEGAL-BERT family use a much smaller legal text collection (11.5 GB) compared with RoBERTa's 161 GB general pre-train corpus, but also achieve competitive results despite less pre-training steps (see Table 3).

Both Harvard Legal-BERT variants present good performance on the ECHR-AM dataset. In contrast to the LEGAL-BERT family which includes ECHR cases as part of its pre-train legal text collection, the pre-train corpus used by Harvard Legal-BERT variants has no overlap with the ECHR-AM dataset. Therefore, for question (b) in Section 1, our case study indicates that domain pre-trained transformers can maintain good generalisability on downstream tasks focusing on different datasets. This "reusable" characteristic of domain pre-trained models is significant. Collecting relevant pre-train corpora and pre-training itself still require sufficient time and computing resources. If the domain pre-trained model is reusable in different domain-specific tasks, sharing domain pre-trained models will be superior to sharing corpora, especially for research groups who do not have the resources or capability for pre-training models on large corpora.

With reference to question (c) in Section 1 about merging domain pre-training with general pre-training, the experiment results indicate that the transformers using further pre-training work slightly better than those using pre-training from scratch. This indicates the potential of enhancing the transformer's generalisability on downstream domain-specific tasks by merging generic corpora

with further pre-training, especially in tasks like argument relation mining that require the model to extract not only the features from the text but also the potential relations between different sequences.

Our work pays attention to a common issue in interdisciplinary NLP research that the background area (i.e., humanities, law) has considerable volumes of text material, but the annotation work is costly and impedes the process of adapting advanced NLP technologies to assist the research. We suggest that pre-train transformers can help this "data poverty" issue, and domain-specific pre-training will improve transformers' performance when adapting to interdisciplinary tasks with only small fine-annotated datasets.

We analyse state-of-the-art transformers in both pre-train categories, and present a comprehensive survey of available models for the legal domain. Our case study provides the first comparison between general pre-trained transformers and domain pre-trained transformers on legal argument mining tasks, and demonstrates that domain pre-trained transformers achieve better results on all three tasks than the baseline outlined by Poudyal et al. (2020). Our case study also compares the performance of two latest groups of transformers in legal domain, and offers an analysis of some key aspects when applying domain pre-training on interdisciplinary NLP tasks.

References

Elena Cabrio and Serena Villata. 2018. Five years of argument mining: a data-driven analysis. In *IJCAI*, volume 18, pages 5427–5433.

Ilias Chalkidis, Ion Androutsopoulos, and Nikolaos Aletras. 2019. Neural legal judgment prediction in english. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4317–4323.

Ilias Chalkidis, Ion Androutsopoulos, and Achilleas Michos. 2018. Obligation and prohibition extraction using hierarchical rnns. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 254–259.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. LEGAL-BERT: The muppets straight out of law school. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2898–2904, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT (1)*.

Emad Elwany, Dave Moore, and Gaurav Oberoi. 2019. Bert goes to law school: Quantifying the competitive advantage of access to large legal corpora in contract understanding. In *Workshop on Document Intelligence at NeurIPS 2019*.

Erwin Filtz, María Navas-Loro, Cristiana Santos, Axel Polleres, and Sabrina Kirrane. 2020. Events matter: Extraction of events from court decisions. In *Legal Knowledge and Information Systems*, pages 33–42. IOS Press.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *EMNLP (Demonstration)*.

John Lawrence and Chris Reed. 2020. Argument mining: A survey. *Computational Linguistics*, 45(4):765–818.

Marco Lippi and Paolo Torroni. 2015. Argument mining: A machine learning perspective. In *International Workshop on Theory and Applications of Formal Argumentation*, pages 163–176. Springer.

Marco Lippi and Paolo Torroni. 2016. Argumentation mining: State of the art and emerging trends. *ACM Transactions on Internet Technology (TOIT)*, 16(2):1–25.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Masha Medvedeva, Michel Vols, and Martijn Wieling. 2020. Using machine learning to predict decisions of the european court of human rights. *Artificial Intelligence and Law*, 28(2):237–266.

Raquel Mochales and Marie-Francine Moens. 2008. Study on the structure of argumentation in case law. In *Legal Knowledge and Information Systems*, pages 11–20. IOS Press.

Raquel Mochales and Marie-Francine Moens. 2011. Argumentation mining. *Artificial Intelligence and Law*, 19(1):1–22.

Ramesh Nallapati and Christopher D Manning. 2008. Legal docket-entry classification: Where machine learning stumbles. In *2008 Conference on Empirical Methods in Natural Language Processing*, page 438.

Prakash Poudyal, Teresa Gonçalves, and Paulo Quaresma. 2019. Using clustering techniques to identify arguments in legal documents. In *ASAIL@ ICAIL*.

Prakash Poudyal, Jaromír Šavelka, Aagje Ieven, Marie Francine Moens, Teresa Gonçalves, and Paulo Quaresma. 2020. Echr: legal corpus for argument mining. In *Proceedings of the 7th Workshop on Argument Mining*, pages 67–75.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Nils Reimers, Benjamin Schiller, Tilman Beck, Johannes Daxenberger, Christian Stab, and Iryna Gurevych. 2019. Classification and clustering of arguments with contextualized word embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 567–578, Florence, Italy. Association for Computational Linguistics.

Mike Schuster and Kaisuke Nakajima. 2012. Japanese and korean voice search. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5149–5152. IEEE.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725.

Milagro Teruel, Cristian Cardellino, Fernando Cardellino, Laura Alonso Alemany, and Serena Villata. 2018. Increasing argument annotation reproducibility by using inter-annotator agreement to improve guidelines. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Douglas Walton. 2009. Argumentation theory: A very short introduction. In *Argumentation in artificial intelligence*, pages 1–22. Springer.

Congcong Wang and David Lillis. 2020. A Comparative Study on Word Embeddings in Deep Learning for Text Classification. In *Proceedings of the 4th International Conference on Natural Language Processing and Information Retrieval (NLPIR 2020)*, Seoul, South Korea.

Lucia Zheng, Neel Guha, Brandon R Anderson, Peter Henderson, and Daniel E Ho. 2021. When does pretraining help? assessing self-supervised learning for law and the casehold dataset. *arXiv preprint arXiv:2104.08671*.

Haoxi Zhong, Chaojun Xiao, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. 2020a. How does nlp benefit legal system: A summary of legal artificial intelligence. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5218–5230.

Haoxi Zhong, Chaojun Xiao, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. 2020b. Jec-qa: A legal-domain question answering dataset. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 9701–9708.

Japanese Beauty Marketing on Social Media: Critical Discourse Analysis Meets NLP

Emily Öhman and **Amy Grace Metcalfe***
Waseda University
ohman@waseda.jp and agmetcalfe@toki.waseda.jp

Abstract

This project is a pilot study intending to combine traditional corpus linguistics, Natural Language Processing, critical discourse analysis, and digital humanities to gain an up-to-date understanding of how beauty is being marketed on social media, specifically Instagram, to followers. We use topic modeling combined with critical discourse analysis and NLP tools for insights into the "Japanese Beauty Myth" and show an overview of the dataset that we make publicly available.

1 Introduction

Instagram is one of the most widely used social media platforms in Japan, with an audience of over 38 million (Clement, 2020). Instagram's focus on photo sharing generates a saturation of images conveying beauty. Since Naomi Wolf's 1991 publication "The Beauty Myth" (Wolf, 1991), countless studies have shown the negative effects the beauty industry has on women, including obsessions over weight (Kayano et al., 2008), anxiety issues (Miller, 2006), and furthering inequality between the sexes (Walter, 2011).

This pilot study examines the linguistic features of Instagram posts by beauty companies, both qualitatively and quantitatively, in order to discover how language contributes to the construction of beauty standards in the Japanese context. We use established natural language processing (NLP) methods and topic modeling, combined with critical discourse analysis (CDA) to understand both the quantifiable data and the social effects of these Instagram posts. This study contributes to a better understanding of the definition of beauty within the Japanese context and offers additional insights into beauty ideals and how these are fabricated. Furthermore, the dataset is made public and will therefore be of use to other researchers as well.

We have chosen to aim our attention at posts made by make up, skincare, hair removal, and hair care companies for the purposes of this study. The reasoning for this choice is due to the the "opt-in" nature of these practices. While other practices such as fashion, or dental care could be considered beautification practices, we have chosen to omit these, as they are either culturally required in the case of clothing (Rouse, 2017) or done primarily for hygiene purposes in the case of dental care. Thus, the companies we have chosen focus only promote practices which are not required for hygiene or protection of modesty.

In the following sections we present the background for our study, explain our data and methods, examine the results and analyze and discuss them in a wider context. We conclude with a discussion including future work related to the dataset.

2 Background

The objectivity of qualitative analyses have been criticized for being too subjective (Cheng et al., 2013). Whether this is entirely justified or not (see e.g. Baškarada and Koronios (2018)), qualitative analyses supported by quantitative methods have proven useful in investigating media discourse, allowing researchers to identify textual patterns (O'Halloran, 2010). Incorporating NLP and topic modeling provides scaffolding for the CDA, leading to verifiable empirical results.

Asian beauty trends have been researched in a multitude of different fields, including but not limited to medicine (Liew et al., 2016), marketing (Li et al., 2008), and sociology and gender studies (Saraswati, 2020). Asia is a diverse market in terms of consumer demographics, culture, and beauty ideals (Yip et al., 2019; Yip, 2018). However, some

[0]Both authors contributed equally to this paper.

Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH), pages 131–137

commonalities can be seen in terms of beauty ideals such as idealization of whiteness and a desire for double eye-lids (Saraswati, 2020).

Although the concept of "whiteness" is not only about idealizing Caucausian skin types, and is traditionally linked to socio-economic status (SES) in Asia, most scholars agree that Western standards for female beauty ideals are prevalent in Asia and include "whiteness" (Jung, 2018).

Surprisingly little work has been done on the topic of beauty by using computational methods. Gender and online slurs have been explored in many papers with the help of NLP tools, but to the best of our knowledge there has not been a study on beauty ideals using methods from even corpus linguistics on modern data. The closest thing we could find was a qualitative study comparing the ideals of female beauty in Malaysian and Belgian advertising (De Cort, 2009), but this study used no NLP methods, and only partially relied on corpora.

In discourse analysis, the topic is well-explored, but relying mostly on qualitative analyses and in some cases linguistic features, Asian beauty ideals are also a common topic (Iqbal et al., 2014; Xu and Tan, 2020; McLoughlin, 2017; Renaldo, 2017).

A related topic is the gendered choice of script in Japanese (Wakabayashi, 2016; Maree, 2013) which is both linked to the producer of the text and the intended audience (Iwahara et al., 2003; Dahlberg-Dodd, 2020; Mellor, 2003). Japanese can be written with three different domestic(-ish) scripts (kanji, hiragana, and katakana). Kanji is usually used for content words, hiragana for syntactic particles and similar but can also be seen as cute and girly (Robertson, 2019), and katakana is used for loan words and emphasis. Additionally the Latin alphabet is also used. Additionally, 和語 (native Japanese words) 漢語 (Sino-Japanese words) and 外来語 (loanwords) can be used to achieve different effects in a similar way to script-choice.

Primarily Fairclough's three-part model (Fairclough, 2001) has been used as a theoretical framework for critical discourse analysis of advertising (see e.g. Kaur et al. (2013) and Lestari (2020)). Consisting of the micro, meso, and macro levels of interpretation, it allows for a comprehensive analysis of texts. At the micro level, linguistic features are identified and investigated. At the meso level, we can see the strategies used and how the message is conveyed, and finally the social context and social effects of such texts will be considered via the macro level. Fairclough's model is particularly well-suited when analyzing social and cultural change, due to the comprehensive nature of the framework (Iqbal et al., 2014).

3 Data

The data was collected using the Instaloader[1] package for Python. The Instagram profiles of twenty companies were randomly chosen, with the criteria being that the company's product or service must be available in Japan and that they have an Instagram profile aimed at the Japanese market. In the cases where the brand had many profiles, the profile mainly written in Japanese was selected. The companies chosen were; Beauteen (beauteen_offical), Chifure (chifure_official), Curél (curel_official_jp), DHC (dhc_official_jp), Etude House (etudejapan), Ichikami (ichikami_kracie) , Innisfree (innisfreejapan), Kanebo (kaneboofficial), Kireimo (kireimo_official), Kosé (kose_official), Liese (liese_official_jp), Maybelline (maybellinejp), Musée (museeplatinum_insta), Palty (paty_official), Revlon (revlonjapan), Rimmel (rimmellondon_jp), Rize Clinic (rizeclinic), Sekkisei (sekkisei.official), Shiseido (shiseido_japan), and TBC Aesthetic (tbc_aesthetic).

This resulted in 7477 posts by these twenty companies, for a total of 365752 lemmas. Depending on the method of adjective inclusion, between 8% and 17% of these were adjectives. The dates of these posts ranged from early 2016 to June 2021. We focused on the text content of the posts and this data is freely available on GitHub[2].

4 Methods

The json-formatted data was then converted to pandas dataframes and flattened for exploratory data analysis. The posts themselves were segmented, tokenized, lemmatized and annotated for part of speech using Fugashi (McCann, 2020), mecab (Kudo, 2006), and spaCy (Honnibal and Montani, 2017). We also used the medium sized Japanese language model for spaCy to improve word recognition.

This mix of general tools and specific tools for Japanese gave us access to both the Western speech tags, such as ADJ - adjectives, but also Japanese tags which made it possible to include 形容詞

[1] https://instaloader.github.io/

[2] https://github.com/esohman/JapaneseBeauty1

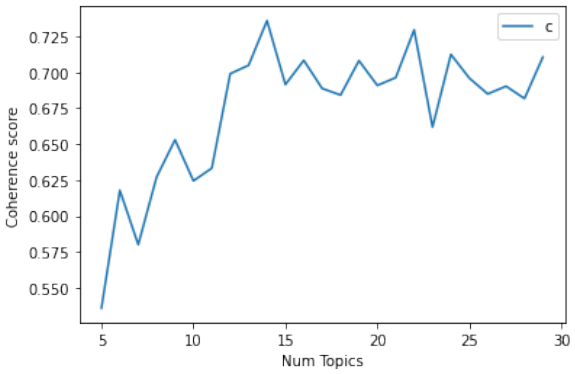

Figure 1: Coherence Values

(adjectives) as well as 形容詞可能 (nominalized adjectives and adjectivized nouns) in our keywords, and therefore not miss important keywords due to overly general linguistic features and part of speech categorization. We chose to focus on adjectives for the linguistic analysis as a quick access point to descriptive language.

The raw text was then cleaned up and used for LDA-based topic modeling using Latent Dirichlet Allocation (LDA) with Mallet as well as NMF-based. It might have made sense to try 20 topics as we had 20 companies, but we assumed many of these companies would be posting somewhat similar content on similar topics and therefore wanted to empirically ascertain the ideal number of topics. We used coherency scores to find the most suitable number of topics for our dataset (see table 1). After 12 topics the coherence scores to sharply rise, topping at 14 with a coherence score over 0.73.

As the dataset is still quite small, we experimented empirically with varying numbers of topics with a few different models (LDA, LDA with Mallet, NMF with Kullback-Leibler and NMF with Frobenius norm). We found that for this data, our most human-interpretable results were achieved by choosing 14 topics with the NMF model using generalized Kullback-Leibler divergence (see figure 2). However, the effect of the different segmentation (spaCy, mecab) and the inclusion of trigrams and bigrams, as well as the use of tfidf vectorizer made the results of these models differ in minor ways but with an impact on the final results. Therefore both the LDA with Mallet model and the NMF with Kullback-Leibler were used for the final analysis with the NMF model getting the clearest and easiest to interpret topics, but with the LDA model finding some interesting underlying topics that were not similarly present in the NMF model.

5 Results

Some of the most representative topics came from the LDA with Mallet model and include those in table 1.

Although the NMF models[3] seemed to be better at homing in on the product types, the LDA models seemed better at finding underlying topics such as mask makeup and self-care, as well as different types of skincare as their own topics.

Another common incursion into the most frequent tokens in topics was brand names. Usernames were stripped from the text, so these occurrences were cases of self-promotion.

We also looked at spelling differences between these adjectives. We looked at the word *kirei* (beautiful, clean) in particular. There were 880 instances of *kirei* in the data of which 52 were of Latin spelling (romaji), 147 were kanji, 83 were hiragana, and 598 were katakana, which also seemed to be favored in hashtags for other words as well.

As can be seen in figure 3, the adjectives and adjectival words have an emphasis on the sensory experience of the products, i.e. how will they make the consumer's skin feel or look like. *Soft, moist, supple, fluffy, smooth, velvety, glowy, dewy* and similar concepts appear alongside *cute* and *just-right*. With the most common adjective(-like) concept being *like/love*.

6 Analysis & Discussion

Overall, patterns that emerge from the corpus, reveal elements of what beauty means in the Japanese context, namely the inclusion of brand equity, and the importance of the skin.The scientific basis for the formulation of components of products seems to also be becoming more common as can be seen in one of the topics (special skincare) in table 1.

Some of the topics that emerged were related to self-care in general (topic 10 in the NMF model). Another topic that emerged from the LDA model was related to COVID-19 and self-care and included words like "mask makeup" (258 occurrences). Most topics were, however, clearly related to cosmetics, skincare, or hair care and included words describing creams, colors, essences, skin tone (yellow-base and blue-base, similar to the terms commonly used in English warm and cool toned). "Whiteness" was a word that came up in

[3]The undecoded characters in Topic 8 are Korean words. We could not find a way to render Japanese and Korean characters simultaneously in a matplotlib plot.

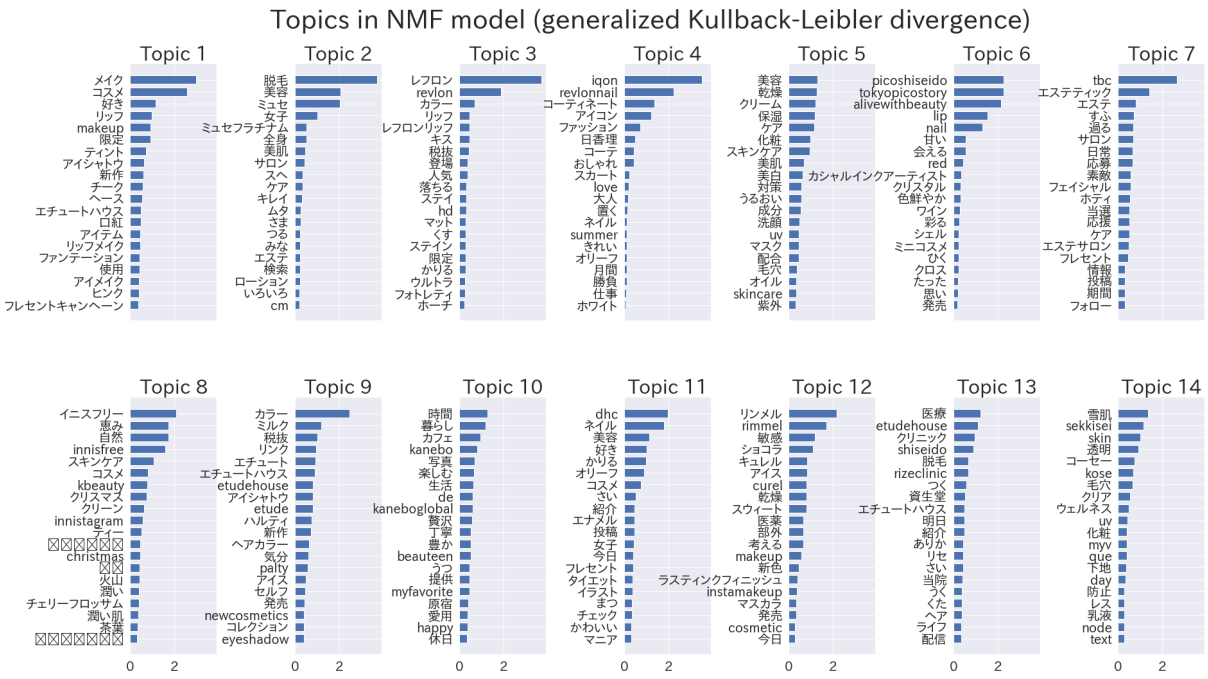

Figure 2: Topics of 14-topic NMF model using Kullback-Leibler divergence

beauty, skin	EN	special skincare	EN	foundation	EN
化粧	make-up	肌	skin	感	feeling
ケア	care	乾燥	dry	毛穴	pore
スキンケア	skin care	敏感	sensitive	肌	skin
クリーム	cream	成分	component	ファンデーション	foundation
美肌	beautiful skin	保湿	moisturizing	ベース	base (loanword)
肌	skin	配合	formulation	下地	base (native JP)
美白	whitening	性	sex, gender	ツヤ肌	dewy skin
UV	UV	picoshiseido		仕上	Finish
美容	Beauty	tokyopicostory		くすみ	Dull

Table 1: Example topics from the LDA with Mallet model

several topics and was not limited to one particular topic, or even subtopic such as skincare. "Whiteness" was most present in the LDA topic, and we believe the reason might be related to the different lemmatizations and segmentation of Japanese (spaCy vs. mecab) as the NMF model seemed to segment the word 美白 as *beautiful* and *white* rather than *whitening*.

Using Fairclough's model to analyze these captions, we can see that certain linguistic features and word choices arise, including a heavy use of adjectives, references to skin issues and solutions, and references to "wellness" culture. Different companies use adjectives differently and these groupings can be seen in both the list of adjectives and adverbs used to describe these ideals as well as in the topics that emerge from topic modeling. Some of the more interesting topics include COVID-related issues such as novel words for mask-specific makeup (i.e. マスクメイク、新商品、美肌作り) and how to take care of yourself and your skin (丁寧な暮らし、暮らしを楽しむ、贅沢な時間、美容好きな人と繋がりたい) during these times.

To analyze the text on the meso level, it is clear that these brands make use of Aristotle's three persuasive strategies (Mooney and Evans, 2018), but in particular ethos, or the credibility of a text. This is achieved through the use of celebrity endorsement which was commonplace among brands. For example, the hair removal clinic TBC have had a long standing partnership with model and personality Rola. When we consider more widely the

Figure 3: Visualization of adjectives and 形容詞、形容詞可能 i.e. adjectives and adjectivized nouns in the corpus

interaction these posts have with their audiences and with society, it gives the impression that these messages are portrayed as being written by someone who cares about your well being, and not a company, thus creating the illusion of friendship. This narrative, paired with the persistent collaborating between brands and celebrities, can contribute to parasocial relationships between brands and their consumers (Yuan et al., 2016).

Notably, the images in the majority of the posts we collected did not actually depict a model or a person at all, and had a greater focus on the product. This is in contrast with billboard advertisements in Japan which typically depict a human model. However, when people were pictured in these social media posts, there was a strong tendency to display only parts of the body, such as a hand, or a typically "beautiful" model (thin, able-bodied, flawless skin etc.). These are the images which may be problematic (Etcoff and Paxton, 2016). The paper explicitly states that self esteem issues increase when women view idealized advertisements, and that has increased since their 2004 report both globally and in Japan (Etcoff and Paxton, 2016). There is strong evidence to suggest that such self esteem issues have real life implications. Kayano et al. (2008) investigation further emphasizes the prominence of eating disorders, abnormal dieting, and an obsession with thinness for both genders among Japanese young people stating that young people are heavily influenced by the media (Kayano et al., 2008)).

It should be noted that the depiction of women in these advertisements differs from the depictions of men whose entire bodies are typically displayed in contrast to women who are often shown in parts. This is a known phenomenon in advertising which perpetuates a "pattern of seeing women as collections of body parts" erasing female identitites (Goldman, 2005; Schroeder and Borgerson, 1998).

7 Conclusions and Future Work

To summarize, we can see that the language and imagery used in Japanese beauty posts on Instagram do reflect modern beauty standards, and make reference to topical issues such as COVID.

We expect this corpus to yield many more interesting insights into Japanese beauty ideals, COVID-related self-care, and societal issues involving the pressure to conform to Japan-specific beauty standards. With future work we hope to dive deeper into the data and specifically we hope to add data produced by consumers themselves. It would be an interesting counterbalance to the marketing speak to be able to compare the content the average consumer outputs with the output of the companies.

Exploring the choice of script might shed more light on the study of loan-word intrusion into Japanese. As script choice is highly gendered in Japanese and there is a lot of promise in this dataset to shed further light on in what contexts the different scripts are used in social media marketing.

We plan on scraping Twitter as well as the comments of YouTube beauty-content producers, as well as personal blogs to achieve this augmentation of our dataset. It would be very interesting to look at the data from a diachronic perspective and see how COVID-19 has affected the consumers' view of beauty products.

References

Saša Baškarada and Andy Koronios. 2018. A philosophical discussion of qualitative, quantitative, and mixed methods research in social science. *Qualitative Research Journal*.

Winnie Cheng et al. 2013. Corpus-based linguistic approaches to critical discourse analysis. *The encyclopedia of applied linguistics*, pages 1353–1360.

Jessica Clement. 2020. Leading countries based on Instagram audience size as of october 2020. *Statista. https://www. statista. com/statistics/578364/countries-with-most-instagram-users/.*

Hannah E Dahlberg-Dodd. 2020. Script variation as audience design: Imagining readership and community in japanese yuri comics. *Language in Society*, 49(3):357–378.

Anne De Cort. 2009. The ideal of female beauty in two different cultures: Socio-cultural analysis of belgian and malaysian print advertisements. *Novitas-ROYAL*, 3(2).

Nancy Etcoff and Susan Paxton. 2016. The Dove global beauty and confidence report.

Norman Fairclough. 2001. *Language and power*. Pearson Education.

Robert Goldman. 2005. *Reading ads socially*. Routledge.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Asma Iqbal, Malik Haqnawaz Danish, and Maria Raja Tahir. 2014. Exploitation of women in beauty products of fair and lovely: A critical discourse analysis study. *International Journal on Studies in English Language and Literature*, 2(9):122–131.

Akihiko Iwahara, Takeshi Hatta, and Aiko Maehara. 2003. The effects of a sense of compatibility between type of script and word in written japanese. *Reading and Writing*, 16(4):377–397.

Jaehee Jung. 2018. Young women's perceptions of traditional and contemporary female beauty ideals in china. *Family and Consumer Sciences Research Journal*, 47(1):56–72.

Kuldip Kaur, Nalini Arumugam, and Norimah Mohamad Yunus. 2013. Beauty product advertisements: A critical discourse analysis. *Asian social science*, 9(3):61.

Mami Kayano, Kazuhiro Yoshiuchi, Samir Al-Adawi, Nonna Viernes, Atsu SS Dorvlo, Hiroaki Kumano, Tomifusa Kuboki, and Akira Akabayashi. 2008. Eating attitudes and body dissatisfaction in adolescents: Cross-cultural study. *Psychiatry and Clinical Neurosciences*, 62(1):17–25.

Taku Kudo. 2006. Mecab: Yet another part-of-speech and morphological analyzer. *http://mecab. sourceforge. jp*.

Eka Marthanty Indah Lestari. 2020. A critical discourse analysis of the advertisement of Japanese beauty products. *IZUMI*, 9(1):58–74.

Eric PH Li, Hyun Jeong Min, and Russell W Belk. 2008. Skin lightening and beauty in four asian cultures. *ACR North American Advances*.

Steven Liew, Woffles TL Wu, Henry H Chan, Wilson WS Ho, Hee-Jin Kim, Greg J Goodman, Peter HL Peng, and John D Rogers. 2016. Consensus on changing trends, attitudes, and concepts of asian beauty. *Aesthetic plastic surgery*, 40(2):193–201.

Claire Maree. 2013. Writing one: Deviant orthography and heternormativity in contemporary japanese lifestyle culture. *Media International Australia*, 147(1):98–110.

Paul McCann. 2020. fugashi, a tool for tokenizing Japanese in Python. *arXiv preprint arXiv:2010.06858*.

Linda McLoughlin. 2017. *A critical discourse analysis of south asian women's magazines: Undercover beauty*. Springer.

Andrew Mellor. 2003. A survey of roman script in written japanese media. *Journal of the Faculty of Policy Management Yokkaichi University*, 2(1_2):101–117.

Laura Miller. 2006. *Beauty up: Exploring contemporary Japanese body aesthetics*. Univ of California Press.

Annabelle Mooney and Betsy Evans. 2018. *Language, society and power: An introduction*. Routledge.

Kieran O'Halloran. 2010. How to use corpus linguistics in the study of media discourse. In *The Routledge handbook of corpus linguistics*, pages 563–577. Routledge.

Zainal Renaldo. 2017. Analysis of linguistic features of beauty product advertisements in cosmopolitan magazine: A critical discourse analysis. *TELL-US Journal*, 3(2):141–54.

Wesley C Robertson. 2019. Why can't i speak in kanji?: Indexing social identities through marked script use in japanese manga. *Discourse, Context & Media*, 30:100297.

Elizabeth Rouse. 2017. Why do people wear clothes? In *Fashion Theory*, pages 122–125. Routledge.

L Ayu Saraswati. 2020. Cosmopolitan whiteness: The effects and affects of skin-whitening advertisements in a transnational women's magazine in indonesia. *Meridians*, 19(S1):363–388.

Jonathan E Schroeder and Janet L Borgerson. 1998. Marketing images of gender: A visual analysis. *Consumption, Markets and Culture*, 2(2):161–201.

Judy Wakabayashi. 2016. Script as a factor in translation. *Journal of World Literature*, 1(2):173–194.

Natasha Walter. 2011. *Living dolls: The return of sexism*. Hachette UK.

Naomi Wolf. 1991. *The beauty myth: How images of beauty are used against women*. Random House.

Huimin Xu and Yunying Tan. 2020. Can beauty advertisements empower women? a critical discourse analysis of the sk-ii's" change destiny" campaign. *Theory and Practice in Language Studies*, 10(2):176–188.

Jeaney Yip, Susan Ainsworth, and Miles Tycho Hugh. 2019. Beyond whiteness: Perspectives on the rise of the pan-asian beauty ideal. In *Race in the Marketplace*, pages 73–85. Springer.

Jesse Wai Chi Yip. 2018. Communicating social support in online self-help groups for anxiety and depression: A qualitative discourse analysis. In *Proceedings of the 32nd Pacific Asia Conference on Language, Information and Computation*, Hong Kong. Association for Computational Linguistics.

Chun Lin Yuan, Juran Kim, and Sang Jin Kim. 2016. Parasocial relationship effects on customer equity in the social media context. *Journal of Business Research*, 69(9):3795–3803.

Text Zoning of Theater Reviews: How Different are Journalistic from Blogger Reviews?

Mylène Maignant[1], Gaëtan Brison[2], Thierry Poibeau[1]

[1]Laboratoire LATTICE (CNRS & ENS-PSL & Université Sorbonne nouvelle)
1, rue Maurice Arnoux, 92120 Montrouge, France
firstname.lastname@ens.psl.eu

[2]Institut Polytechnique de Paris, 5 Av. Le Chatelier, 91764 Palaiseau
gaetan.brison@ip-paris.fr

Abstract

This paper aims at modeling the structure of theater reviews on contemporary London performances by using text zoning. Text zoning consists in tagging sentences so as to reveal text structure. More than 40 000 theater reviews going from 2010 to 2020 were collected to analyze two different types of reception (journalistic vs digital). We present our annotation scheme and the classifiers used to perform the text zoning task, aiming at tagging reviews at the sentence level. We obtain the best results using the random forest algorithm, and show that this approach makes it possible to give a first insight of the similarities and differences between our two subcorpora.

1 Introduction

Since 2010 in England, a wave of blogs written by authors coming from various horizons has arisen on the Internet. Students, theater professionals but also mere amateurs began publishing their own theater reviews. These new independent voices in the digital space progressively redefine the shape of classic journalistic criticism. Although discreet, they offer a new vision of the history of Londonian theaters. By doing so, it sets itself apart from the canon of mainstream journalism.

The emergence of this digital culture triggered a lot of controversies on the status of the review as a literary object. Michael Billington, reviewer for *The Guardian* since 1971, states that a blog 'is more like an informal letter: a review, if it's to have any impact, has to have a definable structure.' For Danielle Tarento, co-founder of the Menier Chocolate Factory, 'a lot of people [bloggers] are not 'proper writers'.' At the other hand of the spectrum, some of these bloggers claim the stylistic singularity of their publications. In the description of Exeuntmagazine.com for instance, the editors claim that: 'Exeunt believes in making beautifully written, *experimental*, fierce and *longform* writing about theatre available for free.'

A review is traditionally organized according to several sections: an introduction, a presentation of the plot, a few lines on the stage, etc. In order to compare the two subcorpora, it is first necessary to segment the reviews into textual zones corresponding to these thematic sections. We assume the two subcorpora will share the same zones, as they are all about theater, but the content of the zones may differ from one subcorpus to the other: e.g., the two communities may not focus on the same aspects of the plays. From a technical point of view, this experiment is also an opportunity to test the robustness and relevance of text zoning across different domains. Text zoning has been mainly used to segment scientific texts so far, but can this technique also be used in the humanities? Can it be used for performance reviews, where critics do not follow a fixed structure, contrary to scientific writing?

This short paper is structured as follows. We first give a brief overview of text zoning. We then present our corpus, the different features and machine learning techniques used for the task. We then comment our results and give some hints on the way these could be used to get a better understanding of the content of the corpus and the differences between the two communities at stake (official critics vs amateur bloggers).

2 Previous Work

The notion of text zoning was first introduced by Simone Teufel in her PhD (Teufel, 1999). Teufel was targeting the automatic analysis of scientific papers. In this context, argumentative zoning refers to the 'rhetorical status of a sentence with respect to the communicative function of the whole paper.' It is for example quite useful to distinguish 'back-

Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH), pages 138–143
December 19, 2021.

ground information' from 'statements of the particular aim of the current paper', to take an example from Teufel's work (Teufel and Moens, 2002). While text zoning has been mainly applied to scientific texts so far, one can also find this technique applied to other domains where it is relevant, for example email messages (Lampert et al., 2009), or job ads (Gnehm and Clematide, 2020).

The number of zones considered varies but is generally around ten or less (7 for example in (Teufel, 1999) and (Guo et al., 2011), or 8 in (Gnehm, 2018)). Zone annotation is generally performed by a group of experts at the sentence level (more rarely at the paragraph level). Inter-annotator agreement on the task is generally high: (Guo et al., 2011) for example reports a score of 0.85 for Cohen's κ (Cohen, 1960).

Once a representative corpus is available, it is possible to train a classifier for the task. Features considered are generally low level (unigrams, bigrams, sometimes specific terms also receive a specific weight) (Teufel and Moens, 2002) but higher level features are also sometimes considered (like syntactic relations in (Guo et al., 2011)). Contextual information (like the previous zone) is also often taken into consideration, since a specific zone tends to appear in typical positions in scientific abstracts. As for training, most recent ML techniques have been explored, from Naive Bayes (Teufel, 1999) to LSTM (Gnehm, 2018), through CRF and SVM (Guo et al., 2011). In this last paper, the authors also investigate semi-supervised learning and active learning, in order to reduce the amount of data needed for training, which often constitutes a bottleneck for the task. More recently, large language models like Bert have also been explored (Gnehm and Clematide, 2020), but they require large amount of data for training.

Here our goal is partly the same as the one in these previous studies. However, our corpus is very different since we analyze theater reviews, which may not be as regular as scientific papers. In our context, zones are important to determine whether the critic is addressing acting, staging or the general setting of the play (we use the rather neutral term 'text zoning' instead of 'argument zoning', since the zones we consider do not always correspond to arguments). Analyzing the overall organization of theater reviews will also make it possible to determine whether these have a rather fixed structure or not, if reviews in newspapers differ a lot from those directly written for blogs on the Web.

3 Corpus Creation

To answer these questions, the first step consisted in collecting the necessary data to create two subcorpora. The first subcorpus is made of journalistic reviews only, while the second one is based on digital theater reviews written by bloggers.

3.1 Subcorpus 1: Journalistic Theater Reviews

The first subcorpus was created thanks to the online database *Theatre Record*. *Theatre Record* is a biweekly paper magazine which reprints in full all the national drama reviews of the productions in London and its regions. Its archives were digitized in 2019 and each newspaper published since 1981 is now available online (in PDF format).

All the newspapers issues have the same characteristics. For each of the shows, a certain number of reviews is given as well as a series of details on the production, such as the cast, the credits and the photographs. The theater in which the play was performed as well as the opening and the closing dates of the show are also indicated. Most of the newspapers represented in this database are well-known among the general public: *The Times, The Guardian, The Independent*, etc. Out of the 84 newspapers available on *Theatre Record*, we have selected 32 of them in total. A number of sources had to be removed. Since this corpus focuses on printed newspapers, online news websites had to be excluded. We also removed newspapers whose reviews were not about London performances and all the newspapers which had a too limited number of reviews.

3.2 Subcorpus 2: Digital Theater Reviews

The second subcorpus is based on 18 English blog platforms whose authors' publications deal with London plays only. The content of these websites was extracted using webscraping techniques. These 18 blog platforms have the following characteristics: they have no printed equivalent, their content is entirely free and their authors are not paid for their activity. They are either run by one person, or by multiple authors.

The selection of these blogs was made according to the top 10 most popular British theater blogs established by Vuelio in 2020. A majority of them also came from the platform *MyTheatreMates*. All

the authors who have their reviews published on *MyTheatreMates* share the following characteristics: They have their own personal website, they post original theatre-related content on their personal website at least once a fortnight, they can provide three professional arts references (e.g. artists they have interviewed or, if they review, producers or publicists who already regularly provide them with complimentary press tickets to shows) and they are active on Twitter.

When this subcorpus was created (September 2020 – April 2021), 52 bloggers were members of *MyTheatreMates*. We selected the blogs which had the highest number of reviews (at least 200 reviews) as well as the ones which were mainly focusing on the Londonian stage.

3.3 Overview of the Corpus

	Newspapers	Blogs
Number of sources	33	18
Number of words	8,831,160	10,364,855
Number of reviews	22781	19045

Table 1: The Descriptive statistics of each corpus (source refers to newspapers vs blog platforms).

Table 1 gives an overview of the two datasets. The corpus is available in textual format (PDFs from *Theatre Record* have been converted and manually corrected) so that NLP tools coming from Stanford could be directly applied. It is to our knowledge the first corpus collecting so many reviews of theater performances. The corpus is freely available online, on the website dedicated to this project: `Dramacritiques.com`.

4 Experiment Description

4.1 Annotation Scheme and Data Labeling

Once the data were collected, the first step consisted in labeling a random sample of reviews that could be used for training. The annotation scheme corresponds to the 8 different possible sections of a review.

The definition of these sections is based on (Fisher, 2015). In his analysis, Fisher examines the various possibilities for one critic to structure his arguments, which leads to the following 8 different categories with 8 different colours:

For this first experiment, the data were labeled by an expert with a strong background in theatre studies. This expert spent more than 15 minutes per review, or 250 hours in total, annotating the sample. Each of the sentences was carefully analyzed to propose the best category it belonged to.

Zone category	Associated colour
Introduction	Purple
Reviewer analysis	Blue
Visual and audio details	Green
Conclusion	Yellow
Performance of actors	Orange
Plot	Red
Structure of the play	Brown
Related to the audience	Grey

Table 2: Delimitation of the different zones and their colours used in the model.

However, some of the sentences could have been classified in two different categories. These cases were recorded and resolved following explicit rules to ensure the consistency of the annotation. 1000 reviews were manually annotated, which was deemed enough for training.

4.2 Data Preparation

Several preprocessing steps were applied to the corpus, following previous experiments in text zoning. Texts were first segmented into sentences, tokenized and tagged (with POS and morphological features) and empty words were removed. Named Entity Recognition and Term Frequency-Inverse Document Frequency (TF-IDF) were also applied on the corpus. Annotations were performed using Stanford tools and were then used as features for training.

In the end, more than eighty variables were created, following previous work in the domain (among others (Teufel, 1999) and (Guo et al., 2011)):

- Statistical variables: average word length, average sentence length, frequency of personal pronouns, etc.
- Tense variables: proportion of verbs in future, present and past tenses
- Grammar variables: top verbs, adjectives, superlatives, nouns, etc.
- Parts of Speech variables: position of the words and their roles in the sentences

- Named Entity Recognition variables: organization, characters, etc.

If the creation of these different types of variables helped to improve the prediction of the algorithms, only a few of them were relevant for the models. We thus applied a feature selection process to reduce the number of variables used during training (we went from more than 100 different features to a little less than 20 main features). We used a correlation matrix, Principal Component Analysis (PCA) to reduce the number of variables we originally had and other tools in function of the models.

4.3 Models for Sentence Classification

We selected 4 traditional classification models that seemed relevant for the task (Naive Bayes, random forest, KNN and RNN). Most of them have already been used for text zoning (see the previous work section), but their relevance in the context of reviews remains to be assessed.

- Naive Bayes is a simple Bayesian model. It is known to perform well on small datasets and will thus constitute a baseline.

- Random Forest (Ho, 1995) is an ensemble learning method that builds a multitude of decision trees at training time. Random forest generally performs better than a single decision tree and can take into account the multiple parameters of our problem.

- K-Nearest Neighbors (KNNs) (Altman, 1992) assumes that all data points in close proximity is labeled with the same class. KNNs may thus not work so well on heterogeneous and diversified data. For this model we tested a wide range of k ranging from 1 to 15 to find the optimal one.

- Recurrent Neural Network (RNN) (Sperduti and Starita, 1997) is relevant to find hidden dependencies patterns in the data. This model is the most powerful one in theory but generally requires more data for training.

We chose to limit ourselves to these well-known classification techniques. More recent approaches exist, for example based on deep learning techniques and using large language models like Bert (Devlin et al., 2019). As we wanted the approach to be portable and easily reproducible by people working in humanities, we excluded these more resource intensive approaches but that is something we should try in the future (see (Gnehm and Clematide, 2020) for an experiment with biLSTM and BERT).

5 Results

We applied each model on the data and computed their accuracy (computed using 10-fold cross validation and averaging the results across folds). Our results are reported in Table 3:

Models	Accuracy
Naive Bayes	.69
Random Forest	.80
K-Nearest Neighbors	.72
Recurrent Neural Networks	.61

Table 3: Performance of the different models.

According to Table 3, the top performing model in terms of accuracy is random forest. The result is comparable to previous studies (for example, (Guo et al., 2011) report .81 overall accuracy).

Note however that we had to find the optimal parameter for the depth of the tree and the number of trees. By doing so the random forest model uses a system of threshold for each important feature (Breiman, 2001). To improve the model and avoid overfitting, we also used cross validation during the training and test steps.

As planned, Naive Bayes is not able to take into account the complexity of the task and performs poorly. KNN also fails at capturing the variations of the different zones, as the texts to classify are quite short. Lastly, RNN performs worse as there are not enough data to train this model properly. For this part in particular, we used Long Short Term Memory Neural Networks which work sequence by sequence. We wanted to use pre-trained models but none of them had already been trained on a similar dataset for this task. The closest we could get was on the IMDB dataset. However, if it were annotated for sentiment analysis, it were not for zoning.

Figure 1 represents a review with the different zones identified, each color corresponding to a specific zone. This figure illustrates how the algorithm works. The model looks for each sentence and calculates its probability of being part of one of the 8 predefined categories. Of course it is possible that one sentence may have different recognizable pat-

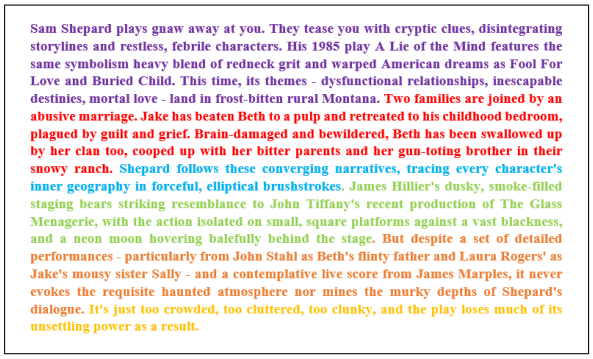

Figure 1: An Example of Annotated Text (each zone is annotated with a specific color). Color code: Purple: Introduction Red: Plot Blue: General Analysis of the Play Green: Visual, Auditory and Audible Details Orange: Actors' Performances Brown: Remarks on the Structure of the Play Yellow: Conclusion

terns that makes it belong to several classes. In this case, the model associates to the sentence the category which has the highest percentage. In function of the class assigned by the model, the sentence will then take the color related to its category.

6 Discussion

Although the accuracy of the algorithm could be improved, these first results are a reliable and relevant base to better understand the comparison between printed and digital theater criticism. If the debate in the artistic sphere highlights the differences between journalists and bloggers, the experiments actually prove that their reviews are more similar than what they claim. Each of the 8 categories we had defined are represented in the two datasets (see Table 4) which suggests that both of them employ similar lines of thought.

Zone category	Newspapers	Blogs
Introduction	15.9	17.0
Reviewer analysis	13.3	11.7
Visual and audio details	4.4	7.3
Conclusion	9.0	8.3
Performance of actors	15.2	18.4
Plot	32.5	28
Structure of the play	8.6	7.2
Related to the audience	0.9	2.1

Table 4: Relative coverage of each predicted zone.

The real differences are located at a subtler level. When we have a closer look at the percentages within each dataset and when we compare them, we can realize that bloggers tend to focus on categories related to affect. 'Visual and audio details', 'Performance of actors' and remarks 'Related to the audience' are all aspects of the review which put to the front the subjective perception of the critic. On the contrary, superior values in percentages for the subcorpus I are situated in categories linked to more factual arguments. 'Reviewer analysis', 'Plot' and 'Structure of the play' rather rely on descriptive and rational materials. This paves the way for further analysis, mixing text zoning and sentiment analysis for example, so as to get a better understanding of the content of the different zones and of the differences between the two corpora under study.

7 Conclusions and Perspectives

We have presented a study based on the automatic analysis of more than 40 000 theater reviews on the contemporary Londonian stage. We have shown that it is possible to segment these reviews into labeled text zones with a good accuracy. In the future, we want to investigate large language models and their potential benefit for the task.

Considering the classification obtained with text zoning, it seems that the two subcorpora considered in the study (journalists vs bloggers) are not as different as some actors of the domain may have claimed, at least from a distant reading perspective. However the content of some zones seems to be really different from one subcorpus to the other: the zoning experiment presented in this paper is thus a first necessary step in order to be able to perform a more precise analysis.

Acknowledgements

Mylène Maignant is partially supported by the EUR (École Universitaire de Recherche) Translitteræ (programme "Investissements d'avenir" ANR-10-IDEX-0001-02 PSL* and ANR-17-EURE-0025). Thierry Poibeau is supported in part by the French government under management of Agence Nationale de la Recherche as part of the "Investissements d'avenir" program, reference ANR19-P3IA-0001 (PRAIRIE 3IA Institute). Lastly, this research also benefited from the support of the CNRS International Research Network Cyclades (Corpora and Computational Linguistics for Digital Humanities).

References

Naomi Altman. 1992. An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, 46:175–185.

Leo Breiman. 2001. Random forests. *Statistics Department University of California Berkeley*.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Mark Fisher. 2015. *How to Write About Theatre*. Methuen Drama, London.

Ann-Sophie Gnehm. 2018. *Text Zoning for Job Advertisements with Bidirectional LSTMs*. University of Zurich.

Ann-Sophie Gnehm and Simon Clematide. 2020. Text zoning and classification for job advertisements in German, French and English. In *Proceedings of the Fourth Workshop on Natural Language Processing and Computational Social Science*, pages 83–93, Online. Association for Computational Linguistics.

Yufan Guo, Anna Korhonen, and Thierry Poibeau. 2011. A Weakly-supervised Approach to Argumentative Zoning of Scientific Documents. In *Empirical Methods in Natural language Processing (EMNLP)*, Edinburgh, United Kingdom.

Tin Kam Ho. 1995. Random decision forests. In *Proceedings of 3rd International Conference on Document Analysis and Recognition*, volume 1, pages 278–282 vol.1.

Andrew Lampert, Robert Dale, and Cécile Paris. 2009. Segmenting email message text into zones. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 919–928, Singapore. Association for Computational Linguistics.

Alessandro Sperduti and Antonina Starita. 1997. Supervised neural networks for the classification of structures. *IEEE Transactions on Neural Networks*, 8/3:714—-735.

Simone Teufel. 1999. *Argumentative Zoning: Information Extraction from Scientific Text*. University of Edinburgh.

Simone Teufel and Marc Moens. 2002. Summarizing scientific articles: Experiments with relevance and rhetorical status. *Computional Linguistics*, 28(4):409–445.

Word Sense Induction with Attentive Context Clustering

Moshe Stekel
Computer Science Dpt.,
Ariel University,
Israel
mstekel@gmail.com

Amos Azaria
Computer Science Dpt.,
Ariel University,
Israel
amos.azaria
@ariel.ac.il

Shai Gordin
Land of Israel Studies
and Archaeology Dpt.,
Ariel University, Israel
shygordin@gmail.com

Abstract

In this paper we present ACCWSI (Attentive Context Clustering WSI), a method for Word Sense Induction, suitable for languages with limited resources. Pretrained on a small corpus and given an ambiguous word (query word) and a set of excerpts that contain it, ACCWSI uses an attention mechanism for generating context-aware embeddings, distinguishing between the different senses assigned to the query word. These embeddings are then clustered to provide groups of main common uses of the query word. We show that ACCWSI performs well on the SemEval-2 2010 WSI task. ACCWSI also demonstrates practical applicability for shedding light on the meanings of ambiguous words in ancient languages, such as Classical Hebrew and Akkadian.

1 Introduction

Natural language expresses human concepts, thoughts, emotions and insights. That is, natural language represents a model of extremely high complexity—the human mind (at least, its communication-driven layers). Some researchers believe that natural language is an environment in which compromise is inevitable when projecting the infinite number of dimensions of human thinking onto the much smaller number of dimensions of human speech (Fedorenko and Varley, 2016). Multiplicity of meaning of a single word, such as polysemy (similarity obtained from a common source) or homonyny (accidental similarity), is therefore an expected product of this compromise. Below are two common examples of word sense ambiguity:

- "I can hear *bass* sounds" versus "They like grilled *bass*"
- "We crossed the river to the other *bank*" versus "Mike deposited the money in his *bank* account"

Humans are able to disambiguate the polysemy/homonyny or understand contextual nuances by using clues that come from the context of the ambiguous word. One of the fundamental tasks of natural language processing is Word Sense Induction (WSI), a task of automatic discrimination of different senses of words by finding these contextual clues.

It is difficult to overestimate the importance of accurate Word Sense Induction when dealing with common Natural Language Processing (NLP) tasks, such as Information Retrieval or Search Clustering. Furthermore, historical research seeks to correctly induce the meaning of words in order to resolve doubts about many historical issues. As a good example we can refer to the Akkadian lemma "galû", the meaning of which ranges between the negative shade of "exile" or "deportation", the neutral shade of "relocation" and the positive one of "appointment". Another example is the Hebrew lemma "zakar", which takes on both the meanings of "memory" and "male". Accurate Word Sense Induction is essential for correct understanding of ancient documents.

In this paper, we present an Attentive Context Clustering WSI (ACCWSI). ACCWSI first creates a word-embedding for each word, which is identical for any context that it appears in. ACCWSI uses the cosine similarity between the words in the context and the word in focus to determine the attention that each word should achieve to form a context aware vector representation for each appearance of the word in focus. ACCWSI then clusters the resulting vectors, such that each cluster represents a different meaning of the word. ACCWSI has demonstrated high practical applicability in languages with limited resources and obtained a very high score by the evaluation framework of SemEval-2 2010 Task 14 (Manandhar et al., 2010). ACCWSI achieved a high score not only with the

Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH), pages 144–151
December 19, 2021.

original training dataset, but also with a training dataset reduced to a fraction of 2.6% of the original dataset, which is comparable to the size of the Hebrew Bible.

2 Related Work

Word Sense Induction and Word Sense Disambiguation provide fertile ground for researchers, starting from very early attempts to tackle these non-trivial tasks, such as "simulated annealing" according to human-edited dictionary (Cowie et al., 1992) and employing the "conceptual distance" between contexts (Agirre and Rigau, 1996), going through later unsupervised methods, that use patterns of word co-occurrence (Bordag, 2006) or bigrams of web search results (Udani et al., 2005), continuing with "hidden concepts" of the contextual words, that not necessarily overlap with the sense of the ambiguous word (Chang et al., 2014), and ending with the most recent solutions like (Eyal et al., 2021), that uses word substitutions of modern Masked Language Models, such as Google BERT MLM.

Our research was inspired by two main works: the context-group discrimination algorithm (Schütze, 1998) from the Context Clustering category and the Google BERT language model (Vaswani et al., 2017). Amrami and Goldberg (Amrami and Goldberg, 2019) utilize Google BERT for their WSI method. However, their method does not meet our requirement of being able to induce word senses in languages with limited resources, as training Google BERT on small corpora does not provide sufficient accuracy (Ezen-Can, 2020). The high scores achieved by the BertWSI model in the SemEval-2 2010 Task 14 (Manandhar et al., 2010) metrics are credited to the fact that the underlying model was pre-trained by Google on a huge corpus of text. Our solution takes advantage of the basic mechanism of attention (Galassi et al., 2020) underlying BERT without applying the complex process of learning attention weights and thus achieves good results when applied to small datasets. The only weight learning process we use is the Word2Vec (Goldberg and Levy, 2014) model training that requires far fewer resources than attention-based learning. Thus, we provide a practical tool in the study of the meanings of words in resource-limited languages, such as ancient dead languages. The Clustering by Committee work (Pantel and Lin, 2002) gave us the idea to use a threshold of 0.5 as an acceptable proportion of orphan instances when measuring the quality of a clustering solution (see Section 3.4.3). We also explored Lin's algorithm (Lin, 1998), which uses the word clustering approach by combining words with similar semantics into sense representations, but it was found less effective when it came to discriminating senses of words in resource-constrained languages.

3 Task and Algorithm

3.1 WSI task definition

The general definition of WSI is automatic detection of the set of senses denoted by a word. A simplified version of WSI can be defined as follows: given a list of lemmatized sentences and a query lemma, find all the sentences in the list that contain the query lemma, and group them so that the instances of the query lemma in one group are semantically similar to each other and noticeably different from the instances in other groups. This is a simplified definition because, when lemmatizing, we ignore some input information, such as the part of speech, tense etc. Note that ignoring the part of speech information of the target word is attractive, especially for ancient genres in which the archaic syntactic forms of words may provide no part of speech information (for instance refer to some hardly explainable verses of the Hebrew Psalms).

3.2 Attention mechanism

Our method uses the following "basic attention" mechanism: given a target word (query) and its "context", either the whole sentence or some "window" of words containing the query word, each element of the context is evaluated by its cosine similarity to the query word. The result is optionally multiplied by a constant factor and eventually softmaxed. We refer to the result as the "weights of similarity" or "weights of relevance". The closer two words are semantically, the greater is the cosine similarity between their embeddings and, therefore, the appropriate weights of relevance are greater. The original word embeddings of the context members are multiplied by the appropriate weights of relevance and thus the power of every context member is improved or worsened according to its relevance to the query word. When these new context-sensitive embeddings are summed into a single vector, this sum represents a context-aware vector of

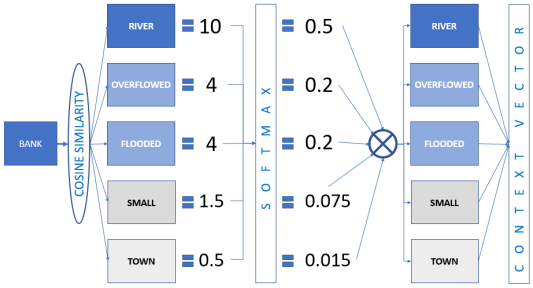

Figure 1: Illustration of the attention mechanism

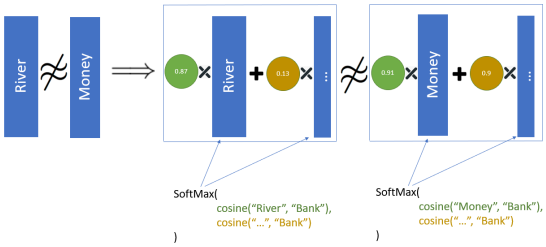

Figure 2: An illustration of separability of context-aware vectors generated by ACCWSI: the most relevant terms (green weights) with respect to the query term "bank" are "river" in the first context and "money" in the second context. They are different and therefore the result context-aware vectors are different. Less relevant terms are multiplied by smaller weights (light brown) and thus have smaller effect on the final context-aware vector.

the query word that embeds its "local sense" with respect to this specific context, where the relevance of each context member is taken into consideration. Figure 1 illustrates this mechanism.

3.3 The ACCWSI algorithm

We now present our Attentive Context Clustering WSI (ACCWSI) algorithm. The ACCWSI algorithm (see Algorithm 1) first replaces the lemmas with their Word2Vec embeddings (Goldberg and Levy, 2014). It then uses the attention mechanism described above (Section 3.2), resulting in context-aware vectors, that are used as input to the DBSCAN clustering algorithm (Schubert et al., 2017), producing clusters of different "shades of meaning" of the query lemma. Since different contexts are best defined by different most relevant context members, and conversely - similar contexts are defined by similar context members, the result vectors can be easily clustered. Figure 2 illustrates this idea.

3.4 Hyperparameters

Algorithm 1 uses several hyperparameters: Word2Vec window, the choice of the clustering algorithm and the internal hyperparameters of the

Algorithm 1 ACCWSI

Input:
text ▷ a list of lemmatized sentences
lemma ▷ a query lemma
Output:
context groups of the query lemma

$model \leftarrow word2vec(text)$
$sentences \leftarrow filter\_by\_lemma(text, lemma)$
$ctx\_aware\_vecs \leftarrow []$
for each $s \in sentences$ **do**
  $ctx\_vecs \leftarrow model.get\_vectors(s)$
  $lemma\_vec \leftarrow model.get\_single\_vector(lemma)$
  $sim \leftarrow cosine\_sim(ctx\_vecs, lemma\_vec)$
  $sim\_weights \leftarrow softmax(sim)$
  $new\_lemma\_vec \leftarrow \sum_i ctx\_vecs_i * sim\_weights_i$
  $ctx\_aware\_vecs.push(new\_lemma\_vec)$
end for
return $DBSCAN(...).fit(ctx\_aware\_vecs)$

latter. The optimal values of these parameters can be found either empirically or by using well-known optimization methods. In this section we explain these hyperparameters, briefly overview the optimization methods, and present the method that achieved best accuracy in our case.

3.4.1 Word2Vec Window

This parameter determines the size of the context to be scanned from each direction around the target word when training the Word2Vec model to perform the missing word prediction task (CBoW architecture) or the context prediction task (Skip-Gram architecture). The optimal value of this parameter intuitively depends on the native average "density of context" inherent to the target language. We found the optimal value empirically by iterating over the range from 2 to 10 and evaluating the result by manually checking the semantic similarity of words suggested by the model. The best values were 5 for English and 2 for Classical Hebrew. This difference is probably due to the specific syntactic structures of Classical Hebrew verses, which are statistically much shorter than the syntactic structures of typical Modern English sentences.

3.4.2 The choice of the clustering algorithm

We evaluated several different clustering algorithms on our task, including KMeans (Hamerly and Elkan, 2004), Gaussian-Mixture model (Reynolds, 2009) and DBSCAN (Schubert et al., 2017). DBSCAN performed slightly better and was therefore selected as our clustering algorithm.

3.4.3 DBSCAN-eps

This parameter is a key one for the density-based clustering proposed by DBSCAN. It defines the maximum distance between two points to be considered as neighbors. There are several methods in the literature for optimizing the value of this parameter, such as the Kneedle algorithm for finding the maximum curvature in the graph of distances, the Silhouette Score for evaluating the clustering quality, and more. Although these optimization methods demonstrated good performance (unsupervised V-Measure of 15.3%), we propose a heuristic that performed better. The rationale behind the heuristic is that text can contain instances of ambiguous words with highly clear context, in addition to other instances with more obscure context. Decreasing the value of eps results in clearer but tighter clusters, filtering out distant "noisy" instances. In our case, narrowing the clusters while keeping the number of the "noisy" instances below 50% gave good results. Algorithm 2 demonstrates this heuristic.

Algorithm 2 Fine-tuning the DBSCAN eps hyperparameter - the value of eps is iteratively decreased until the noise (the fraction of the orphan instances) becomes greater than $\frac{1}{2}$

```
best_eps ← 0.95
for each x ∈ range(90, 0, −5) do
    eps ← x/100
    labels ← DBSCAN(eps = eps)
      .fit(cxt_aware_vectors)
    noise ← labels.count(−1)/len(labels)
    if noise ≤ 0.5 then
        best_eps ← eps
    else
        break
    end if
end for
return best_eps
```

4 Experimental evaluation

We ran an experiment to evaluate the algorithm on **Sem-Eval 2010 Task 14** (Manandhar et al., 2010), which aims to objectively measure and compare the quality of WSI systems. Both training and test data are English sentences containing polysemous or homonymous nouns and verbs. The goal of the task is to split the instances of each ambiguous word and their contexts into clusters representing different meanings. The result is assessed by comparison with the "Gold Standard" clustering performed by human experts. In Section 4.1 we present the Unsupervised V-Measure and F-Score metrics of this assessment as well as the Supervised Recall metric.

4.1 SemEval-2 2010 Task 14 Evaluation

In Task 14 of the SemEval-2 2010 workshop (Manandhar et al., 2010), participants were asked to train their models on the corpus of training data provided by the organizers, and then perform word sense induction for a set of sentences containing both ambiguous nouns and ambiguous verbs. The results were assessed against the "Gold Standard" clusters compiled by human experts. The tables below show the metrics achieved with ACCWSI trained on the full training corpus (by training ACCWSI we mean training its internal Word2Vec model), as well as the metrics achieved with the reduced ACCWSI, which was trained on a randomly selected 2.6% of the training data, along with those of the participants with the highest scores in every metric.

System	VM % (All)	VM % (Nouns)	VM % (Verbs)
ACCWSI full	17.3	20.7	12.3
Hermit	16.2	16.7	15.6
UoY	15.7	20.6	8.5
KSU KDD	15.7	18	12.4
ACCWSI reduced	15.4	18.8	10.4
Duluth-WSI	9	11.4	5.7
...			
...			
Duluth-WSI-SVD-Gap	0	0	0.1

Table 1: V-Measure (VM) unsupervised evaluation. V-Measure assesses the quality of a clustering solution by explicitly measuring its homogeneity and its completeness. Homogeneity refers to the degree that each cluster consists of data points primarily belonging to a single Gold Standard class, while completeness refers to the degree that each Gold Standard class consists of data points primarily assigned to a single cluster. V-Measure is the harmonic mean of the homogeneity and completeness.

System	FS % (All)	FS % (Nouns)	FS % (Verbs)
Duluth-WSI-SVD-Gap	63.3	57	72.4
KCDC-PT	61.8	57	72.4
...			
...			
ACCWSI reduced	55.9	51.3	62.7
...			
...			
ACCWSI full	53.8	47.2	63.4
Duluth-WSI-SVD	41.1	37.1	48.2
Duluth-WSI	41.1	37.1	48.2
...			
...			
Duluth-R-110	16.1	15.8	16.4

Table 2: Paired F-Score (FS) unsupervised evaluation: two sets of instance pairs are generated - a set of all possible instance pairs within each induced cluster and a set of all possible instance pairs within each Gold Standard class. Precision is the number of common instance pairs between the two sets to the total number of pairs in the induced clusters, while recall is the number of common instance pairs between the two sets to the total number of pairs in the Gold Standard classes. F-Score is the harmonic mean between precision and recall.

System	SR % (All)	SR % (Nouns)	SR % (Verbs)
ACCWSI full	63.7	59.6	71.1
ACCWSI reduced	62.7	57.5	69.8
UoY	62.4	59.4	66.8
Duluth-WSI	60.5	54.7	68.9
...			
...			
Duluth-Mix-Uni-Gap	18.7	1.6	43.8

Table 3: Supervised recall (SR) using a test set split with 80% mapping and 20% evaluation. In this evaluation, the testing dataset is split into a mapping and an evaluation corpus. The first one is used to map the automatically induced clusters to Gold Standard senses, while the second one is used to evaluate methods in a WSD setting.

5 Application examples

In this section we present examples of applying our method to a relatively small Hebrew corpus—the Hebrew Bible. We used the text-fabric version of the BHSA project to generate the appropriate dataset and run the ACCWSI algorithm on it. Figure 3 shows the operation of the ACCWSI algorithm used to obtain two different meanings of "bank" in English. Figure 4 and Figure 5 present the induced classes for two ambiguous Hebrew Biblical lemmas: **khalal** (dead body/desecrate) and **zakar** (male/memory). The instances of the first lemma were split into 2 sense clusters while the instances of the second lemma were split into 5 sense clusters. ACCWSI seems to perform well and provide satisfactory clusters despite the small training corpus.

6 Future work

Iterating the process of generating context embeddings may improve the accuracy of the clustering. In our future work we plan to develop a method for determining the "center of mass" (or "centroid" for convex clusters) of every cluster. These centers will be treated as new "query" embeddings and the ACCWSI attention-weighted technique will be reapplied within each cluster using its new query (its center). This should provide finer discrimination of meanings. This iterative process can be repeated many times until maximum accuracy is achieved.

Another effort we lead these days is Word Sense Induction in ancient Akkadian texts. Between the 9th to the late 7th centuries BCE, the Assyrian Empire deported millions of people across the Near East. By even the most humble estimates, around 1.3 million people were moved around as a result of conquest, labour recruitment or as punishment, just to name the central reasons for this dire process (Sano 2020). However, the records for these deportations are numerous and came down to us in different genres that deal with the act of deportation, or forced migration, from different points of view: contemporaneous Assyrian royal inscriptions, letters and administrative texts, as well as Babylonian historical chronicles, written many years after the events in question. All were written in Assyrian and Babylonian, two close dialects of Akkadian, the oldest known (East-)Semitic language in the world. In all, 19 different verbs deal with various stages of the forced migration, like the capture of people or forced recruitment, their change of location, and resettlement. Even then, there are differences across meanings for specific verbs, sometimes minute ones, but also quite substantial in terms of semantics.

A good example of such a complicated verb is *galû* which the Chicago Assyrian Dictionary (CAD), the most comprehensive dictionary of Akkadian, translates as "1. to go into exile, 2. to deport, to exile (Š-stem, causative)" (CAD Š/3, 201). Its usage is limited to a Babylonian context, either in Assyrian letters dealing with Babylonia or Babylonian chronicles (Sano 2020, 34). As text 1 shows, the usage, much like that of Biblical Hebrew *GLY/H*, is used in consequence of a military

Sentence	Attention Highlights	Cluster
Her bank account was rarely over two hundred.	account, rarely	0
After breakfast, she closed her account at the bank and turned in her resignation.	account, close, turn	0
How could a man with four million in the bank be in financial danger?	financial, man, danger	0
Seating herself on a low bank, she studied the souls.	seat, study, low	0
If you would know the history of these homesteads, inquire at the bank where they are mortgaged.	mortgage, homestead, history	0
I guess he had some bucks at one time – back when he bought all this land – but his bank account never held a candle to mine.	account, hold, buy	0
A stream bank is the terrain alongside the bed of a stream	stream, stream, bed	1
He walked up and down the river, leading his hourse behind him; but he kept his eyes turned always toward the dim, dark spot which he knew was the old North Church.	river. church, spot	1
She waded to the bank and picked up he shoes and stockings.	stocking, shoe, wade	1
The town of Barwani is situated near the left bank of the Nerbudda	town, near, left	1
Cushing himself swam to the swamps on the river bank, and after wading among them for hours reached a Federal picket boat.	river, boat, swamp	1
Within an hour, there were riding side-by-side down the south bank of the creek, searching for the blocked area.	creek, area, south	1

Figure 3: Two different meanings of **bank**, the financial institute and the geographic terrain, are represented by the clusters in the figure. The "attention highlight" column shows the most relevant context words. The first cluster contains an interesting failure: the fourth sentence is clustered as a financial institute even though a human would cluster it as a geographic terrain. The reason is that the most relevant context words "seat, study, low" are not sufficiently indicative

.

Sentence	Attention Highlights	Cluster
וּמִזַּרְעֲךָ לֹא־תִתֵּן לְהַעֲבִיר לַמֹּלֶךְ וְלֹא תְחַלֵּל אֶת־שֵׁם אֱלֹהֶיךָ אֲנִי יְ־הוָ־ה.	נתן, י־ה־ו־ה, עבר	0
לָכֵן אֱמֹר לְבֵית יִשְׂרָאֵל כֹּה אָמַר אֲדֹנָי יְ־הוִ־ה לֹא לְמַעַנְכֶם אֲנִי עֹשֶׂה בֵּית יִשְׂרָאֵל כִּי אִם לְשֵׁם קָדְשִׁי אֲשֶׁר חִלַּלְתֶּם בַּגּוֹיִם אֲשֶׁר בָּאתֶם שָׁם.	גוי, ישראל, קדש	0
וְאֶחְמֹל עַל שֵׁם קָדְשִׁי אֲשֶׁר חִלְּלוּהוּ בֵּית יִשְׂרָאֵל בַּגּוֹיִם אֲשֶׁר בָּאוּ שָׁמָּה.	חמל, גוי, ישראל	0
וּבוֹא אֶל הַגּוֹיִם אֲשֶׁר בָּאוּ שָׁם וְיַחֲללוּ אֶת שֵׁם קָדְשִׁי בֶּאֱמֹר לָהֶם עַם יְ־הוָ־ה אֵלֶּה וּמֵאַרְצוֹ יָצָאוּ.	עם, גוי, קדש	0
וְנָפְלוּ חֲלָלִים בְּאֶרֶץ כַּשְׂדִּים וּמְדֻקָּרִים בְּחוּצוֹתֶיהָ.	דקר, חוץ, נפל	1
כִּי נָתַתִּי אֶת חִתִּיתִי בְּאֶרֶץ חַיִּים וְהֻשְׁכַּב בְּתוֹךְ עֲרֵלִים אֶת חַלְלֵי חֶרֶב פַּרְעֹה וְכָל הֲמוֹנָה נְאֻם אֲדֹנָי יְ־הוִ־ה.	ערל, חרב, נאם	1
אוֹתָם יִרְאֶה פַּרְעֹה וְנִחַם עַל כָּל הֲמוֹנוֹ חַלְלֵי חֶרֶב פַּרְעֹה וְכָל חֵילוֹ נְאֻם אֲדֹנָי יְהוִה.	חרב, נאם, כל	1
שָׂמָה נְסִיכֵי צָפוֹן כֻּלָּם וְכָל צִדֹנִי אֲשֶׁר יָרַד אֶת חֲלָלִים בָּחִתִּיתָם מִגְבוּרָתָם בּוֹשִׁים וַיִּשְׁכְּבוּ עֲרֵלִים אֶת חַלְלֵי חֶרֶב וַיִּשְׂאוּ כְלִמָּתָם אֶת יוֹרְדֵי בוֹר.	ערל, כל, כל	1

Figure 4: In the Hebrew Bible, the lemma **khalal** normally takes on the sense of either **dead body**(as a noun) or **desecrate**(as a verb). This figure presents the appropriate clusters generated by ACCWSI. The "attention highlight" column shows the most relevant context words. In the context of **desecrate** (cluster 0), the attention is paid to words like **God**, **sacred**, **nation** etc. while in the context of **dead body** (cluster 1), the attention is paid to **sword**, **stab**, **fall** etc.

conflict. However, a single instance in a letter from the time of Tiglath-pileser III (c. 731-730 BCE), here text 2, shows that under certain political circumstances people could ask for someone to deport them to Assyria, perhaps referring to the safety of being a protected refugee under the direct responsibility of the Assyrian king. This might also be the meaning of certain cases in Aramaic, where *gly* in G-stem active participle means "exile, refugee", or in D-stem means "to emigrate", (Comprehensive Aramaic Lexicon, s.v. gly D and C).

Text 1: SAA 19, 27 rev. 4'-8a' (online edition, Luukko 2012) 4' LUGAL? *lu*? *ú-di* NIM.MA.KI_-a_-[*a*] 5' LÚ.ERIM-MEŠ-*šú-nu* TA DUMU ^{m}GIN—NUMUN *la*? 6' *i-du-ku ù ša—da-a-ni* 7' *ú-sag-li-šú-nu šú-nu-ú-ma* 8' *ig-da-al-ú*

Sentence	Attention Highlights	Cluster
וּבֶן שְׁמֹנַת יָמִים יִמּוֹל לָכֶם כָּל זָכָר לְדֹרֹתֵיכֶם יְלִיד בַּיִת וּמִקְנַת כֶּסֶף מִכֹּל בֶּן נֵכָר אֲשֶׁר לֹא מִזַּרְעֲךָ הוּא.	דור, יום, כל	0
פְּקֻדֵיהֶם בְּמִסְפַּר כָּל זָכָר מִבֶּן חֹדֶשׁ וָמָעְלָה פְּקֻדֵיהֶם שִׁבְעַת אֲלָפִים וְחָמֵשׁ מֵאוֹת.	מספר, שבע, פקד	0
בְּמִסְפַּר כָּל זָכָר מִבֶּן חֹדֶשׁ וָמָעְלָה שִׁמֻנַת אֲלָפִים וְשֵׁשׁ מֵאוֹת שֹׁמְרֵי מִשְׁמֶרֶת הַקֹּדֶשׁ.	מספר, מעל, כל	0
וּפְקֻדֵיהֶם בְּמִסְפַּר כָּל זָכָר מִבֶּן חֹדֶשׁ וָמָעְלָה שֵׁשֶׁת אֲלָפִים וּמֵאתָיִם.	מספר, פקד, מעל	0
זְכָר רַחֲמֶיךָ יְ-הֹ-וָ-ה וַחֲסָדֶיךָ כִּי מֵעוֹלָם הֵמָּה.	עולם, חסד, רחמים	1
חַטֹּאות נְעוּרַי וּפְשָׁעַי אַל תִּזְכֹּר כְּחַסְדְּךָ זְכָר לִי אַתָּה לְמַעַן טוּבְךָ יְ-הֹ-וָ-ה.	פשע, חסד, נעורים	1
פְּנֵי יְהוָה בְּעֹשֵׂי רָע לְהַכְרִית מֵאֶרֶץ זִכְרָם.	כרת, ה', רע	1
אֱלֹהִים עֲלֵי נַפְשִׁי תִּשְׁתּוֹחַח עַל כֵּן אֶזְכָּרְךָ מֵאֶרֶץ יַרְדֵּן וְהַרְמֹנִים מֵהַר מִצְעָר	אלהים, נפש, כן	1
וְאִם מִן הַצֹּאן קָרְבָּנוֹ לְזֶבַח שְׁלָמִים לַי-הֹ-וָ-ה זָכָר אוֹ נְקֵבָה תָּמִים יַקְרִיבֶנּוּ.	נקבה, תמים, ה'	2
אוֹ הוֹדַע אֵלָיו חַטָּאתוֹ אֲשֶׁר חָטָא בָּהּ וְהֵבִיא אֶת קָרְבָּנוֹ שְׂעִיר עִזִּים זָכָר תָּמִים.	חטא, תמים, קרבן	2
לְהַעֲלֹת תָּמִים זָכָר בַּבָּקָר בַּכְּשָׂבִים וּבָעִזִּים.	תמים, רצון, כשב	2
וְאִם זֶבַח שְׁלָמִים קָרְבָּנוֹ אִם מִן הַבָּקָר הוּא מַקְרִיב אִם זָכָר אִם נְקֵבָה תָּמִים יַקְרִיבֶנּוּ לִפְנֵי יְהוָה.	נקבה, תמים, ה'	2
וַיֵּצֵא אֲלִיקִים בֶּן חִלְקִיָּהוּ אֲשֶׁר עַל הַבַּיִת וְשֶׁבְנָא הַסּוֹפֵר וְיוֹאָח בֶּן אָסָף הַמַּזְכִּיר אֶל חִזְקִיָּהוּ קְרוּעֵי בְגָדִים וַיַּגִּידוּ לוֹ אֶת דִּבְרֵי רַב שָׁקֵה.	ספר, בן, אסף	3
וַיֵּצֵא אֲלִיקִים בֶּן חִלְקִיָּהוּ אֲשֶׁר עַל הַבַּיִת וְשֶׁבְנָא הַסֹּפֵר וְיוֹאָח בֶּן אָסָף הַמַּזְכִּיר.	ספר, בן, אסף	3
אֶלְיָקִים וְאַחְיָה בְּנֵי שִׁשָּׁא סֹפְרִים יְהוֹשָׁפָט בֶּן אֲחִילוּד הַמַּזְכִּיר.	ספר, בן, יהושפט	3
וַיִּקְרָאוּ אֶל הַמֶּלֶךְ וַיֵּצֵא אֲלֵהֶם אֶלְיָקִים בֶּן חִלְקִיָּהוּ אֲשֶׁר עַל הַבַּיִת וְשֶׁבְנָה הַסֹּפֵר וְיוֹאָח בֶּן אָסָף	ספר, בן, אסף	3
וְהַיָּמִים הָאֵלֶּה נִזְכָּרִים וְנַעֲשִׂים בְּכָל דּוֹר וָדוֹר מִשְׁפָּחָה וּמִשְׁפָּחָה מְדִינָה וּמְדִינָה וְעִיר וָעִיר וִימֵי הַפּוּרִים הָאֵלֶּה לֹא יַעַבְרוּ מִתּוֹךְ הַיְּהוּדִים וְזִכְרָם לֹא יָסוּף מִזַּרְעָם.	דור, דור, יום	4
יְ-הֹ-וָ-ה שִׁמְךָ לְעוֹלָם יְ-הֹ-וָ-ה זִכְרְךָ לְדֹר וָדֹר	דור, דור, עולם	4
אֲזַכִּירָה שִׁמְךָ בְּכָל דֹּר וָדֹר עַל כֵּן עַמִּים יְהוֹדוּךָ לְעוֹלָם וָעֶד.	דור, דור, שם	4
וְאַתָּה יְהוָה לְעוֹלָם תֵּשֵׁב וְזִכְרְךָ לְדֹר וָדֹר.	דור, דור, עולם	4

Figure 5: In the Hebrew Bible, the senses of the lemma **zakar** are related to either **male** or **memory**. This figure presents the five clusters generated by ACCWSI. The "attention highlight" column shows the most relevant context words. The first cluster represents the sense of **male human**, the second one - **God's memory**, the third one - **male animal sacrifice**, the fourth - **the role of scribe** and the fifth - **chronological memory**

(rev. 4'-5') The Elamites killed their soldiers with the son of Mukin-zeri and (6'-8') **deported them by force**. They too **went into exile**.

Text 2: SAA 19, 87 obv. 8b'-13a' (online edition, Luukko 2012) 8' *. . . e-gir-tum ša ina* UGU 9' [md]AMAR.UTU—A—SUM-*na na-u-ni-ni it-tab-lu-ni* 10' [*ina*] *pa-ni-ni i-si-si-ú* : *ù* m*ba-la-su* 11' [*ip*]-*ta-la-a a—da-niš ma-a an-nu-rig* x+[x x] 12' [*at*]-*tu-nu tal-la-ka ma-a ša-ga-la-ni* [o] 13' [*i*]-*si-ku-nu la-al-lik* . . .

(obv. 8'-9') They intercepted the letter which was brought to Merodach-baladan (10') and read it [in] our [pr]esence. But Balassu (11') [g]ot very scared, saying: (12') "You (pl.) must come this moment and **deport me!** (13') I will go [wit]h you (pl.)."

7 Conclusion

In this paper we propose ACCWSI, an algorithm to automatically induce various senses of ambiguous words by automatically focusing on the most relevant words from their contexts. After learning generic word embeddings into a Word2Vec model, ACCWSI uses the basic attention technique for determining the most relevant context members and generating context-aware embeddings, each with a semantic direction that aggregates the directions of its context members. Distant meanings imply distant context embeddings and vice versa, and thus standard clustering techniques can be easily applied for grouping the context embeddings by their common semantic directions. ACCWSI has shown excellent performance even when trained on a small subset of the training data in the SemEval-2 2010 task 14. Furtheremore, ACCWSI demonstrated high applicability in disambiguation of word senses in ancient Semitic languages, such as Classical Hebrew and Akkadian.

Acklowledgments

This research is supported by the Ministry of Science & Technology, Israel, Grant 3-16464.

References

Eneko Agirre and German Rigau. 1996. Word sense disambiguation using conceptual density. *arXiv preprint cmp-lg/9606007*.

Asaf Amrami and Yoav Goldberg. 2019. Towards better substitution-based word sense induction. *arXiv preprint arXiv:1905.12598*.

Stefan Bordag. 2006. Word sense induction: Triplet-based clustering and automatic evaluation. In *11th Conference of the European Chapter of the Association for Computational Linguistics*.

Baobao Chang, Wenzhe Pei, and Miaohong Chen. 2014. Inducing word sense with automatically learned hidden concepts. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 355–364.

Jim Cowie, Joe Guthrie, and Louise Guthrie. 1992. Lexical disambiguation using simulated annealing. In *COLING 1992 Volume 1: The 14th International Conference on Computational Linguistics*.

Matan Eyal, Shoval Sadde, Hillel Taub-Tabib, and Yoav Goldberg. 2021. Large scale substitution-based word sense induction. *arXiv preprint arXiv:2110.07681*.

Aysu Ezen-Can. 2020. A comparison of lstm and bert for small corpus. *arXiv preprint arXiv:2009.05451*.

Evelina Fedorenko and Rosemary Varley. 2016. Language and thought are not the same thing: evidence from neuroimaging and neurological patients. *Annals of the New York Academy of Sciences*, 1369(1):132.

Andrea Galassi, Marco Lippi, and Paolo Torroni. 2020. Attention in natural language processing. *IEEE Transactions on Neural Networks and Learning Systems*.

Yoav Goldberg and Omer Levy. 2014. word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *arXiv preprint arXiv:1402.3722*.

Greg Hamerly and Charles Elkan. 2004. Learning the k in k-means. *Advances in neural information processing systems*, 16:281–288.

Dekang Lin. 1998. Automatic retrieval and clustering of similar words. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 2*, pages 768–774.

Suresh Manandhar, Ioannis Klapaftis, Dmitriy Dligach, and Sameer Pradhan. 2010. Semeval-2010 task 14: Word sense induction &disambiguation. In *Proceedings of the 5th international workshop on semantic evaluation*, pages 63–68.

Patrick Pantel and Dekang Lin. 2002. Discovering word senses from text. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 613–619.

Douglas A Reynolds. 2009. Gaussian mixture models. *Encyclopedia of biometrics*, 741:659–663.

Erich Schubert, Jörg Sander, Martin Ester, Hans Peter Kriegel, and Xiaowei Xu. 2017. Dbscan revisited, revisited: why and how you should (still) use dbscan. *ACM Transactions on Database Systems (TODS)*, 42(3):1–21.

Hinrich Schütze. 1998. Automatic word sense discrimination. *Computational linguistics*, 24(1):97–123.

Goldee Udani, Shachi Dave, Anthony Davis, and Tim Sibley. 2005. Noun sense induction using web search results. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 657–658.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Transferring Modern Named Entity Recognition to the Historical Domain: How to Take the Step?

Baptiste Blouin[1,2], Benoit Favre[1], Jeremy Auguste[2], and Christian Henriot[2]

[1]Aix Marseille Univ, Université de Toulon, CNRS, LIS, Marseille, France
firstname.lastname@lis-lab.fr
[2]Aix Marseille Univ, Institut de Recherches Asiatiques, Aix, France
firstname.lastname@univ-amu.fr

Abstract

Named entity recognition is of high interest to digital humanities, in particular when mining historical documents. Although the task is mature in the field of NLP, results of contemporary models are not satisfactory on challenging documents corresponding to out-of-domain genres, noisy OCR output, or old-variants of the target language. In this paper we study how model transfer methods, in the context of the aforementioned challenges, can improve historical named entity recognition according to how much effort is allocated to describing the target data, manually annotating small amounts of texts, or matching pre-training resources. In particular, we explore the situation where the class labels, as well as the quality of the documents to be processed, are different in the source and target domains. We perform extensive experiments with the transformer architecture on the LitBank and HIPE historical datasets, with different annotation schemes and character-level noise. They show that annotating 250 sentences can recover 93% of the full-data performance when models are pre-trained, that the choice of self-supervised and target-task pre-training data is crucial in the zero-shot setting, and that OCR errors can be handled by simulating noise on pre-training data and resorting to recent character-aware transformers.

1 Introduction

Due to the massive effort to digitize and transcribe historical documents, the field of digital humanities is facing the challenges of digesting and analyzing large quantities of texts.

With the continuous advancements of natural language processing (NLP), there is a growing interest in applying tools such as named entity recognition (NER) on historical documents. Indeed, identifying people, places, and other historical entities is crucial to understand the historical context, and having the ability to do so automatically is a major step forward.

	Models	LitBank	HIPE
Off the shelf	Spacy, en_core_web_sm	21.27	12.35
	Spacy, en_core_web_trf	28.36	19.60
	Stanford CoreNLP	23.59	31.23
SOTA	Boros et al.	–	63.20
	Ju et al.	68.30*	–
Ours	Zero-shot	70.44	13.73
	Full	81.53	62.32

Table 1: Off-the-shelf NER performance on historical texts from the LitBank and HIPE test sets in a zero-shot setting and performance of State-Of-The-Art systems trained on target data. Reported values are F1-scores. * denotes a somewhat different experimental setting, which makes this result incomparable.

It facilitates the exploration of massive corpora by identifying, counting and extracting textual clues, among others to enrich a database, which can help to systematically explore the information reported in these documents. But the variety present in historical texts, compared to modern ones, makes the evaluation and application of NLP techniques quite difficult. In particular, apart from the fact that these documents relate to different domains, the evolution of language, as well as the noise due to optical character recognition (OCR) errors, preclude using off-the-shelf systems.

NER success, like for other NLP tasks, is highly dependent on the corpus on which the system has been trained, and most of the available named entity (NE) corpora use contemporary texts with contemporary concerns. For example, off-the-shelf NER systems such as SpaCy (Honnibal et al., 2020) or Stanford's CRF-NER (Manning et al., 2014) do not yield acceptable results on historical texts as evidenced in Table 1. Therefore, domain adaptation and the zero-shot setting are very relevant to applying NLP on historical documents. In this study, zero-shot setting denotes the training of a

Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH), pages 152–162
December 19, 2021.

system on contemporary data and an evaluation on historical data.

The goal of this study is to evaluate the effort required to obtain relevant NER results on historical documents. This effort can relate to annotation in the target domain, transfer of contemporary models, pre-training on matching resources, or adaptation to OCR errors. Compared to other domain transfer approaches (Jia et al., 2019; Liu et al., 2020; Sachan et al., 2018), where the evaluation is carried out on specific contemporary domains, this work considers NER from a historian's point of view: we wish to process historical texts and understand why some approaches do not yield usable results. In that context, would off-the-shelf systems be appropriate? If not, can we reduce the amount of annotation needed to obtain reasonable results by using existing resources? And finally, if so, is this approach robust to the characteristic difficulties of historical documents?

Our contribution is to tentatively answer the following questions: (1) What annotation effort in the target historical domain? (2) Is it worth adapting initial pre-trained word representations? (3) What is the impact of OCR errors on transfer performance? We perform experiments on the LitBank (Bamman et al., 2019) and HIPE (Ehrmann et al., 2020) annotated historical NER datasets using prototypical NER systems built on the previous and current generation of models, trained on contemporary annotations from ACE 2005 (Walker et al., 2006) and various amounts of target data.

2 Related work

NER is a typical sequence labeling task where the aim is to locate and classify named entities mentioned in unstructured text into pre-defined categories such as person names, organizations, locations, etc. The research around this task has allowed obtaining very good results on modern documents. For example, on the CoNLL-03 dataset the results reach up to 94.9% of F1-score. From the numerous existing approaches, we describe a few representative recent works. Wang et al. (2020b) propose an Automated Concatenation of Embeddings to build adequate system input representations for structured predictions. It will calculate error based on results of the training process and then compare it with other combinations to finally find out the most suitable concatenation embedding layer for the problem. Yamada et al. (2020) propose a model that treats words and entities in a given text as independent tokens, and outputs contextualized representations of them. Their Transformer based language model pre-trained on both large-scale text corpora and knowledge graphs achieves SOTA performance on various entity related tasks. Wang et al. (2021) propose to use external contexts to improve model performance by retrieving and selecting a set of semantically relevant texts through a search engine and constructed a new representation through the concatenation of a sentence and its external contexts.

Given the progress on this task and its need for applications, many off-the-shelf models pre-trained on modern data are made available. Spacy (Honnibal et al., 2020), Flair (Akbik et al., 2019), Stanza (Qi et al., 2020), AllenNLP (Gardner et al., 2018) offer models trained on OntoNotes 5.0, Stanford CoreNLP (Manning et al., 2014) provides a model trained on a mixture of CoNLL, MUC-6, MUC-7 and ACE.

In this paper we mainly focus on NER in English but this task is also a subject of research in other languages. For instance, Cao et al. (2018) proposed an adversarial transfer learning strategy to make full use of the boundary information shared by tasks and prevent the task-specific functions of Chinese word segmentation. Rahimi et al. (2019) process 41 languages using truth inference to model the transfer annotation bias from diverse source-languages models. Xie et al. (2018) created a 0-shot NER systems by aligning monolingual embeddings from English to Spanish, German, and Dutch, and then translated the English CoNLL dataset into these languages, and built a self-attentive Bi-LSTM-CRF model using the translated languages.

Cross domain NER Cross-domain approaches have been developed to enhance the generalization of NER models to a given target domain.

Most existing approaches are in a supervised setting where both source and target domains have labeled data, the goal being to improve performance over only using instances from the target data. Baselines jointly train models on source and target data with shared parameters, add adaptation layers on top of source models for capturing target domain specifics, or design label-aware feature representations for NER adaptation (Daumé III, 2007; Wang et al., 2018a).

More specific methods use multi-task approaches which have been shown to be effective for

this cross-domain task, to reduce the gap between the two domains. For example, Jia et al. (2019) propose to use extrinsic data in both the source and target domains to train language models for domain adaptation. Wang et al. (2018b) propose a parameter transfer learning between feature representations from Bi-LSTM and two conditional random fields. Wang et al. (2020a) propose a multi-task learning objective that learns domain labels as an auxiliary task and Zhou et al. (2019) propose a Dual Adversarial Transfer Network which aims at addressing representation difference and resource data imbalance problems.

Methods such as transfer learning also show that knowledge sharing is effective for cross-NER. For instance, Lee et al. (2018) utilize transfer learning by initializing a target model with parameters learned from source-domain NER, and rely on labeled target domain data to finetune the model. Cao et al. (2018) propose an adversarial transfer learning framework for Chinese NER task, which can exploit task-shared word boundaries features and ensure proper information usage from the word segmentation task. Lin and Lu (2018) perform adaptation across two domains using adaptation layers augmented on top of the existing neural model. Yang et al. (2019) propose a fine-grained knowledge fusion model to balance the learning from the target data and learning from the source model.

NER on historical texts Given the number of general NER papers produced over the last few decades in the NLP field, studies targeting historical texts and literary documents are still scarce.

In general, research on this subject have not only experimented with NER applied to historical materials but also many of them have addressed some of the most pressing challenges involved in the use of current state-of-the-art NER systems on historical texts: disparate quality of digitization and OCR, handling of non-European or classical languages, or dealing with spelling variations. Packer et al. (2010) experimented with recognition of person names in noisy OCR texts using a Dictionary-Based, Regex-Based, Maximum Entropy Markov and CRF models, and evaluated the output against hand-labelled test data. Grover et al. (2008) built a rule-based NER system for recognizing names of places and persons in digitized records by focusing on issues caused by the high level of variance in the use of word-initial upper-case letters, as well as issues connected to the use of OCR technology. Rodriguez et al. (2012) evaluated four tools for NER on historical texts including OpenNLP (Kwartler, 2017) and Stanford NER. They showed that the Stanford NER system had the overall best performance. Rodrigues Alves et al. (2018) show that character-level word embedding, combined with a Bi-LSTM-CRF model, can help reduce the impact of OCR errors and handle rare words in 19-21C scholarly books and journals. More recent approaches (Schweter and März, 2020) evaluated the impact of word embeddings at the level of their learning and their combination, on this task. Labusch et al. (2019) apply a model based on multilingual BERT embeddings, which is further pre-trained on large OCRed historical German unlabelled data and subsequently finetuned on several NER datasets. They show that an appropriately pre-trained BERT model delivers decent performance in a variety of settings. Boros et al. (2020) added a two task-specific transformer layers on top of the pre-trained BERT to alleviate data sparsity issues. However, the use of recent word representations, such as BERT, is not totally suitable, as its ability to handle noisy data remains a point to be clarified as to its robustness (Sun et al., 2020).

Compared to them, we do not seek to optimize the performance on specific historical data, but rather propose a replicable transfer procedure linking the effort to be provided on the target domain in order to have performance relative to those obtained on contemporary data.

3 Datasets

In this study, we deal with target domain annotated datasets (historical texts), and source domain annotated datasets (contemporary texts, typically news). Table 2 outlines dataset statistics for both domains.

Two target datasets are used, each with two different subdomains, specific difficulties towards this task and a non-comparable annotation guideline.

LitBank (Bamman et al., 2019) is an annotated dataset of 100 English-language literary public-domain texts from Project Gutenberg, annotated with ACE entity categories except for the weapon category (person, location, geopolitical entity (GPE), facility, organization (ORG), and vehicle). In contrast to existing datasets built primarily on news (focused on GPEs and ORGs), literary texts offer strikingly different distributions of entity categories, with much stronger emphasis on people and description of settings. All texts were

Domain	Dataset	Train	Dev	Test	NE freq.	Types	Text sources	Time Period
Target	LitBank	29,894	4,133	3,425	18%	7	Novels	1852-1923
	HIPE	0	2,575	1,301	9%	5	Newspapers	1798-2018
Source	ACE 2005	34,669	4,336	3,777	24%	7	News, speech, web	2003-2004

Table 2: Datasets statistics. **train/dev/test** columns represent the number of named entities. **NE freq.** represents the ratio between the number of entities and the number of words. **Types** indicates the number of different entity categories.

published before 1923, with the majority falling between 1852 and 1911.

HIPE (Ehrmann et al., 2020) is a collection of digitized documents covering three different languages: English, French, and German. The documents come from archives of several Swiss, Luxembourgish, and American newspapers. The corpus was manually annotated by native speakers according to the HIPE impresso guidelines, which are derived from the Quaero[1] annotation guide. The corpus is annotated with 5 types of entities: person, location, organization, time and production. The time-span of the whole corpus goes from 1798 to 2018. The particularity of this dataset is that it contains OCR errors, with no gold alignment. Feuilleton, tabular data, crosswords, weather forecasts, time schedules and obituaries were excluded as well as articles that were fully illegible due to OCR errors. In this study, only the English part of this corpus is used: annotations are only available for development and testing, but not for training.

One source dataset is used. **ACE 2005** (Walker et al., 2006) Multilingual Training Corpus was developed by the Linguistic Data Consortium (LDC) and contains approximately 1,800 documents of mixed genre texts in English, Arabic, and Chinese annotated for entities, relations, and events. The genres include newswire, broadcast news, broadcast conversation, blog, discussion forums, and conversational telephone speech. The dataset is annotated with 7 entity types: person, location, GPE, facility, organization, vehicle, and weapon. We followed the same pre-processing of the data as those presented by Bamman et al. (2019).

In general, all corpora share annotations in persons, locations and organizations, these three types also represent the majority of annotations. ACE and LitBank share 100% of their entity types and ACE and HIPE share 3 out of 6 entity types, which represents 92.5% of entity instances of shared type in the test set. A shared entity type is an entity type with the same name (e.g., Person).

4 Experimental Settings

We used three different systems which represent the typical kind of systems that could be implemented in an industry-provided solution for historical NER given current technology.

BERT (Devlin et al., 2019), with an extra layer to predict NER categories. The pre-trained contextual embeddings are finetuned on the source/target dataset using their proposed approach. It's a *de facto* baseline for NLP systems, and is implemented with the transformers library (Wolf et al., 2020). We also used BERT models (**BERT-Hist**) (Hosseini et al., 2021) finetuned on 47,685 books (5.1B tokens) in English from the year 1760 to 1900 from the Microsoft British Library Corpus. This model, compared to BERT-Base learned from Wikipedia and Bookcorpus, allows us to question the impact of diachronic language changes on these embeddings applied to the historical domain.

CharBERT (Ma et al., 2020) which accounts for characters in addition to BPE tokens. The model is pre-trained with a Noisy LM objective for obtaining robust character-level representations. We expect such model to perform well on noisy OCR, and rely on the available implementation[2].

LSTM-CRF (Lample et al., 2016) initialized with FastText (Bojanowski et al., 2016) non-contextual embeddings, an architecture that has shown its robustness for NER and is standard in many available systems. This allows to compare its performance to contextual ones. The Flair library (Akbik et al., 2019) is used for this model.

All results presented in the experiments section are averages over 10 random initializations. Since the objective is not to maximize performance on a particular dataset or on a particular architecture, the same hyperparameters for all experiments of a given system are used.

The learning rate is initialized to 0.1 for the LSTM-CRF and to 3e-5 for the transformers. For

[1] http://www.quaero.org/media/files/bibliographie/quaero-guide-annotation-2011.pdf

[2] https://github.com/wtma/CharBERT

the three architectures, the size of the hidden layers are set to 512 and the training is performed on 3 epochs on the source data (10 epochs for the LSTM-CRF) and finetuned on 10 epochs on the target data. The rest of the parameters are the defaults proposed by the libraries for the different models. NER performance is computed with F1-score (F1). Because not all corpora contain nested entities, the embedded entity mention is removed and the smaller mention is kept in the case where the datasets proposed this type of entity. Moreover, as previously mentioned, the HIPE dataset do not share all entity types. The evaluation of a system is only performed on the entity types of the test set, and all the predicted entities that are not part of the test set tagset are removed prior to evaluation.

5 Experiments

5.1 What annotation effort in the target domain?

The low performance of off-the-shelf systems, exemplified in Table 1, suggests that adaptation is necessary.

First, systems are evaluated according to the amount of annotated data in the target domain. For this experiment, the ACE English NER dataset is considered as the source domain, and LitBank and HIPE as the target domain. LitBank originates from the same annotation guidelines as ACE while HIPE is based on different guidelines. We first pretrain the different systems on the source domain data, then we finetune them on the target domain samples with a varying quantity of inputs, from 10 sentences to the maximum number of sentences available in the target corpus, up to 6k sentences for LitBank, which is already a large number of annotated sentences. We compare this approach to a system trained only on the target data. This experiment allows evaluating the expected performance given the annotation effort in the target domain, and outlines the importance of pre-training. The results of this experiment are given in Table 3.

First, when the systems are already pre-trained, depending on the amount of target data used, the results, independently of the system used, vary from 17.7% to 27.3% of F1 using 10 sentences and from 46.8% to 61.1% of F1 using the whole HIPE corpus. In the case where our systems were only trained on the HIPE dataset, the three systems obtain 0% F1 for a training on 10 sentences and can vary from 39.1% to 57.6% using the whole corpus.

When pre-trained on ACE data, the three systems present similar score evolutions according to the amount of data used (modulo the maximum value obtained by each system). Only 10 sentences of the target dataset are required to achieve performance that is more than one third of the maximum performance that could be achieved using the entire dataset. Compared to training only on 10 sentences, without pre-training, where the systems fail to learn. By using 50 sentences from the target dataset one can obtain more than two thirds of the maximum performance, so compared to training without pre-training on source data, 50 sentences is still insufficient to generalize on the test set, except for CharBERT, which in this case manages to get more than half of the maximum performance. Above this quantity of annotated sentences, the finetuning approach presents a constant improvement while keeping results superior to training from scratch on the target data. However, from 250 annotated sentences BERT and LSTM-CRF have enough data to learn without ACE data, at this stage, these two systems can get about 75% of the maximum performance. Concerning CharBERT, it recovers 95% of the maximum performance using about half of the available data.

The results obtained on LitBank are similar to those observed on HIPE except that in the case of finetuning from ACE, the target dataset shares the same annotation guideline as the source and doesn't contain noise, which explains the high performance even with low amounts of target data.

However, since this dataset provides more annotated sentences, the analysis can be taken further. At 400 sentences on LitBank, in the case of a pre-training on ACE, systems are within 92-96% of the maximum performance when using 6000 sentences. But in the case of training from scratch on the target, performance is well below with 86%, 79% and 61% of the maximum F1 for CharBERT, BERT and LSTM-CRF respectively. When using 6000 sentences, except for LSTM-CRF, not using pre-training gives the same results as when using it. In practice, having 6000 annotated sentences already requires a big annotation effort. In these results, transfer approaches show that they require fewer annotated sentences than training from scratch from a more realistic amount of 1000 annotated sentences. Indeed, for BERT and LSTM-CRF, in the case of pre-training on ACE and with the addition of 250 annotated sentences on the target corpus, better

Models / Splits		LitBank							HIPE				
		10	50	250	400	1000	3000	6000	10	50	250	400	444
Pre-trained	CharBERT	68.1	71.0	75.1	76.0	77.6	79.8	80.6	27.3	46.9	57.9	61.1	61.1
	BERT	69.4	73.3	75.9	76.8	78.9	80.7	80.9	25.2	47.8	54.6	57.4	58.1
	LSTM-CRF	55.9	62.4	67.6	69.4	71.5	74.7	75.5	17.7	31.6	44.1	46.8	46.8
Only	CharBERT	00.0	30.5	64.0	69.2	74.5	78.8	80.1	00.0	33.8	54.2	57.3	57.6
	BERT	00.0	00.0	54.3	63.7	72.9	79.0	80.4	00.0	00.0	42.6	51.5	52.1
	LSTM-CRF	00.0	00.0	24.5	44.4	57.8	68.3	72.9	00.0	00.4	29.6	37.6	39.1

Table 3: F1 obtained on the targets test set depending on the system used as well as the amount of training on the target used in the case where our systems are already **pre-trained** on ACE and in the case where our systems were **only** trained on the target.

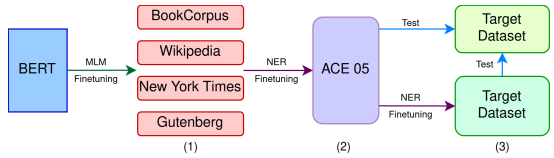

Figure 1: A three-step procedure: (1) BERT representations are first finetuned on two million sentences from unannotated corpora (BookCorpus fiction, Wikipedia, New York Times newspaper, or Gutenberg fiction), (2) the representations are further finetuned to train an ACE NER system thanks to an added decision layer, and (3) that model is finally finetuned on the target NER annotated dataset.

performances are obtained (75.9% and 67.6% of F1 respectively) than using models trained from scratch on 1000 sentences (72.9% and 57.8% of F1 respectively). CharBERT requires 400 annotated sentences when it is pre-trained on ACE to obtain better performance than a system learned only on 1000 sentences from LitBank. This increase can be explained by the fact that CharBERT requires less data than other systems to learn on new data. Indeed, 50 annotated sentences allows obtaining 30.5% of F1 with CharBERT compared to 0% with the other systems.

Due to the different annotation guidelines, the type of data and the quality of the documents used, cross-NER to a historical domain requires at least some annotation in the target domain. Nevertheless, we could see that pre-training a system on contemporary data allows to considerably decrease the amount of annotation needed. Through these experiments two thresholds are observed, the first one at 250 sentences, which allows obtaining very promising results on a distant domain on a low budget. The second, estimated at 1000 sentences, allows obtaining almost optimal performance.

5.2 Is it worth adapting initial word representations?

A realistic scenario for digital humanities is to have access to large annotated corpora in the target domain. However, reaching the scale of data required for pre-training a BERT-like language model is unlikely in the target domain and training such models from scratch is still computationally expensive. Jia et al. (2019) show that finetuning contextual embeddings with pre-training task on a relatively small amount of unannotated texts can improve transfer results. The approach, called domain adaptive pre-training (DAPT), consists in adapting word representations prior to training and transferring the NLP task at hand. Therefore, we evaluate the impact of finetuning the BERT representations to the target texts prior to training the NER system compared to using a BERT trained on historical data (Hosseini et al., 2021). The experimental procedure is described in Figure 1.

The four pre-training datasets have been selected for the following reasons. **Perfect match**: Gutenberg is the source corpus for LitBank, it represents the perfect adaptation corpus due to its proximity to the target. **Genre match, time mismatch**: The New York Times corpus represents a partial match since, like HIPE, it is a newspaper corpus, but it is not from the same period nor from the same source. **Similar genre and time**: By finetuning the representations on Bookcorpus we want to focus the representations on its literary side in order to observe the improvement obtained on LitBank, without totally modifying the distribution following the addition of new data. **Complete mismatch**: Wikipedia does not share anything with the target but is a general domain corpus.

As a comparison, we add randomly initialized BERT baselines to question the importance of the match between the pre-training domain and the source and target domains. Finally, we finetuned a

Pre-training	LitBank		
	Only	*0-shot*	*Full*
No init baseline	34.00	10.71	34.02
BERT-Base	80.40	**70.44**	80.80
BERT-Hist	79.76	63.91	80.09
Book Corpus	79.60	66.66	79.63
Wikipedia	79.83	67.01	80.20
New York Times	78.69	65.13	79.75
Gutenberg	**81.03**	64.64	**81.53**
LitBank-NER	80.53	70.31	80.41

Table 4: NER F1 performance obtained on the LitBank test sets according the unannotated corpus on which BERT pre-training tasks are finetuned. A zero-shot scenario (only trained ACE), a full training scenario (trained ACE then finetuned on target data) and in the case where our systems were **only** trained on the target are compared.

Pre-training	HIPE		
	Only	*0-shot*	*Full*
No init baseline	08.34	01.72	08.44
BERT-Base	52.10	13.44	58.14
BERT-Hist	**60.96**	09.61	**58.34**
Book Corpus	51.63	11.60	53.51
Wikipedia	52.98	**13.73**	56.19
New York Times	51.26	11.96	54.88
Gutenberg	56.41	12.97	57.86
HIPE-NER	53.50	09.84	55.14

Table 5: NER F1 performance obtained on the HIPE test sets according the unannotated corpus on which BERT pre-training tasks are finetuned. A zero-shot scenario (only trained ACE), a full training scenario (trained ACE then finetuned on target data) and in the case where our systems were **only** trained on the target are compared.

BERT only on the NER training data of the target.

The results of this experiment (table 4 for LitBank and table 5 for HIPE) show that finetuning word representations has a small, often negative, impact on NER performance. In the full training scenario, using DAPT improves NER in the LitBank case (+0.78 compared to BERT) by using matching data (Gutenberg). Surprisingly, that same data corresponds to the worst results for zero-shot on LitBank (-5.79 points). On HIPE, where no target training data are available, BERT-base is the most stable in the transfer scenarios and performance is very low in the 0-shot setting across the board, due to mismatch in annotation guidelines. In addition, for this data set, being distant in annotation and domain from our source data, learning using only the target data with historical embeddings leads to better results (60.96).

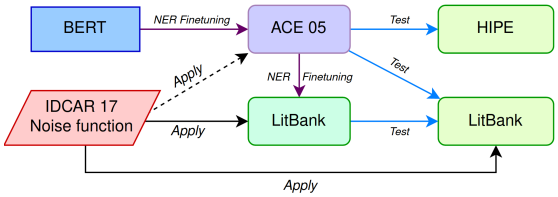

Figure 2: Procedure used to evaluate the impact of noise on the transfer models. The dotted arrow corresponds to the second part of the experiments.

5.3 What is the impact of OCR errors on transfer performance?

Historical texts are often the result of digitization and OCR from the original paper documents. This process leads to inevitable errors due to the quality of the source material and the variability of the print or handwriting.

The last experiment presented in this paper evaluates the impact of OCR errors on NER. Given the target corpora, it's difficult to evaluate their real impact. On the one hand the HIPE texts contain OCR errors, but we do not have manually corrected texts, and on the other hand, the LitBank texts are OCR-error free. Therefore, we chose to simulate an OCR system by randomly adding errors to LitBank.

Following previous work (Pruthi et al., 2019), we change the original character sequence by deleting, inserting, and substituting characters within it. Figure 2 illustrates the procedure that is used in order to mimic OCR errors and to evaluate their influence on the NER task. In order to obtain a realistic error distribution, the error probabilities are computed from the ICDAR 17 corpus, which contains both OCR from several systems and reference texts. This approach allows to control the amount of noise applied to the dataset, simulating various OCR difficulty settings. As a first step, transfer quality from regular ACE to noisy LitBank is evaluated with BERT, BERT-Hist and CharBERT systems, the assumption being that the latter shall cope better with errors. The results as a function of the amount of noise injected into the target corpus are shown in table 6. In a second step, the same transfer is evaluated after corrupting the source data (ACE) using the same process. We varied the amount of noise from 0% to 10% calculated as the character error rate. Results are presented in tables 7 and 8.

Obviously, the two systems are sensitive to noise, and as the amount of noise increases, F1 decreases. First, it can be seen in Table 6 that combining a

Noise	Only			0-shot			Full		
	BERT-Hist	*BERT*	*CharBERT*	*BERT-Hist*	*BERT*	*CharBERT*	*BERT-Hist*	*BERT*	*CharBERT*
0%	79.74	80.40	80.10	63.91	70.44	67.38	80.08	80.80	80.61
1%	79.01	78.46	79.46	64.37	67.73	66.17	79.09	78.96	79.92
2.5%	76.28	75.69	77.72	61.61	64.03	63.86	76.78	76.45	78.14
5%	73.27	71.46	75.36	58.29	57.81	60.58	73.93	71.97	75.79
10%	67.00	63.83	70.52	47.71	47.76	48.50	67.87	65.20	70.39

Table 6: Results obtained on the LitBank test depending on the amount of noise injected in the **target** corpus in a **0-shot** scenario (trained on ACE), in a **full** training scenario (trained on ACE, finetuned on LitBank) and when our systems were **only** trained on LitBank.

Noise	0-shot		
	BERT-Hist	*BERT*	*CharBERT*
0%	65.67	70.44	67.38
1%	64.95	68.03	66.76
2.5%	62.46	65.23	65.52
5%	59.15	60.73	62.25
10%	52.80	51.70	57.63

Table 7: Results obtained on the LitBank test depending on the amount of noise injected in the **source and target** corpora in a **0-shot** scenario (trained on ACE).

Noise	Full		
	BERT-Hist	*BERT*	*CharBERT*
0%	80.08	80.80	80.61
1%	79.37	78.69	79.54
2.5%	76.88	76.06	77.60
5%	73.72	72.14	75.50
10%	68.33	65.34	70.69

Table 8: Results obtained on the LitBank test depending on the amount of noise injected in the **source and target** corpora in a **full** training scenario.

system learned on clean data with noisy test data, CharBERT is less sensitive to noise compared to BERT the more degraded the data are. And the same goes for BERT-Hist which was learned from text created by OCR. In the case where we evaluate only the systems learned on ACE in a zero-shot scenario, the two systems show different behavior regarding noise. When the target data are noisy at 10%, we observe a strong drop of the F1, due to a drop in recall for BERT and BERT-Hist and to a drop in precision for CharBERT. In general, the results show that CharBERT is more robust to noise but suffers from the same performance degradation as BERT when only trained on clean data, but evaluated on noisy data. However, when a model first trained on clean data is finetuned with noisy texts, CharBERT's performance is much better than BERT's with a difference of 5.19 points of F1 at the 10% noise level. This difference, reduced to 2.52 points of F1 with BERT-Hist.

Compared to the previous results, when the same noise distribution is applied to the source dataset (Tables 7 and 8), performance is similar when finetuning on the noisy target dataset. However, in the case where the models trained on noisy ACE are directly evaluated on noisy LitBank, a large improvement is observed compared to when ACE is not noisy. Indeed, recall for the BERT models improves compared to when the source data was clean. The same behavior is observed for CharBERT, where also the recall improves.

However, in the zero-shot case, BERT learned as well as CharBERT up to 2.5% noise. Above that, the noisier the data are, the more robust CharBERT is compared to BERT, with a performance improvement of 5.93 points of F1 with 10% noise. Reduced to 4.83 points of F1 with BERT-Hist.

In view of the LitBank results with simulated noise, a similar approach can be applied for the source data when processing the HIPE corpus. We finetune CharBERT on ACE with 10% noise from the IDCAR 17 distribution and then finetune this model on HIPE which has its own natural OCR error distribution. The results show that training on noisy data, even though the noise follows a different error distribution, improves the results (+1.22 points) on a noisy dataset. However, in the zero-shot scenario, we have a drop of 0.56 points.

6 Discussion

We identify replicable steps for the application of NER on historical documents. First, select contemporary resources, pre-trained models or annotated datasets, using a guideline close to the target needs. Then annotate a few sentences on the target data to adapt the learned models on the source data. 250 sentences can recover 93% of the full-data performance and 1000 sentences 98%. In a second step, the choice of the model will depend on the quality of the target documents to process. If the target is noisy, CharBERT, even compared to a BERT learned on historical noisy data, is more robust to process this type of documents. Moreover, if the

quality information of the target documents is available, applying noise following the same distribution to the source documents will allow the system to be more robust on the target data. Finally, DAPT can be applied if a large amount of unannotated target data is available ($>$ 2M sentences) and if some target data is annotated. This approach does not seem to work in the case where no annotated target data is available. In the case of 0-shot, using word embeddings adapted to the source data will bring better performance on the target data.

7 Conclusion

In this work, we investigate the potential transfer of contemporary named entity recognition models to the historical domain.

Experiments show that finetuning contemporary pre-trained transformers allows reducing considerably the annotation effort and can be further reduced by making an informed choice of the data sources for transfer. Adapting pre-trained word representations prior to learning the task (DAPT) allows a low-cost adaptation to the target domain and improves performance in the full settings depending on the dataset. Processing noisy data is still challenging but the choice of an architecture relying on pre-trained character representations, and the simulation of target noise on the source domain allows recovering acceptable performance compared to a BERT baseline.

To further this study, we would like to systematically look into recent transformer architectures and pre-train them on large corpora of historical texts instead of crawled web data. It would be an opportunity to infuse the models with knowledge of the target temporal span. In addition, we would like to study how levels of NER performance impact historian's findings, and whether current technology is acceptable for reliably mining large quantities of historical documents.

Acknowledgements

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 788476).

This work was granted access to the HPC resources of IDRIS under the allocation 2021-AD011012594 made by GENCI.

References

Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. 2019. Flair: An easy-to-use framework for state-of-the-art nlp. In *NAACL 2019, 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59.

David Bamman, Sejal Popat, and Sheng Shen. 2019. An annotated dataset of literary entities. In *Proceedings of the 2019 Conference of the North*, pages 2138–2144, Minneapolis, Minnesota. Association for Computational Linguistics.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomás Mikolov. 2016. Enriching word vectors with subword information. *CoRR*, abs/1607.04606.

Emanuela Boros, Elvys Linhares Pontes, Luis Adrián Cabrera-Diego, Ahmed Hamdi, José G. Moreno, Nicolas Sidère, and Antoine Doucet. 2020. Robust Named Entity Recognition and Linking on Historical Multilingual Documents. In *Conference and Labs of the Evaluation Forum (CLEF 2020)*, volume 2696 of *Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum*, pages 1–17, Thessaloniki, Greece. CEUR-WS Working Notes.

Pengfei Cao, Yubo Chen, Kang Liu, Jun Zhao, and Shengping Liu. 2018. Adversarial Transfer Learning for Chinese Named Entity Recognition with Self-Attention Mechanism. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 182–192, Brussels, Belgium. Association for Computational Linguistics.

Hal Daumé III. 2007. Frustratingly easy domain adaptation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 256–263, Prague, Czech Republic. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Maud Ehrmann, Matteo Romanello, Alex Flückiger, and Simon Clematide. 2020. Extended Overview of CLEF HIPE 2020: Named Entity Processing on Historical Newspapers. In *CLEF 2020 Working Notes. Conference and Labs of the Evaluation Forum*, volume 2696. CEUR-WS. Number: CONF.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. AllenNLP: A deep semantic natural language processing platform. In *Proceedings of Workshop for*

NLP Open Source Software (NLP-OSS), pages 1–6, Melbourne, Australia. Association for Computational Linguistics.

Claire Grover, Sharon Givon, Richard Tobin, and Julian Ball. 2008. Named entity recognition for digitised historical texts. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Kasra Hosseini, Kaspar Beelen, Giovanni Colavizza, and Mariona Coll Ardanuy. 2021. Neural language models for nineteenth-century english. *CoRR*, abs/2105.11321.

Chen Jia, Xiaobo Liang, and Yue Zhang. 2019. Cross-Domain NER using Cross-Domain Language Modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2464–2474, Florence, Italy. Association for Computational Linguistics.

Meizhi Ju, Makoto Miwa, and Sophia Ananiadou. 2018. A neural layered model for nested named entity recognition. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1446–1459, New Orleans, Louisiana. Association for Computational Linguistics.

Ted Kwartler. 2017. *The OpenNLP Project*, chapter 8. John Wiley and Sons, Ltd.

Kai Labusch, Clemens Neudecker, and David Zellhöfer. 2019. Bert for named entity recognition in contemporary and historic german. In *KONVENS*.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural Architectures for Named Entity Recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270, San Diego, California. Association for Computational Linguistics.

Ji Young Lee, Franck Dernoncourt, and Peter Szolovits. 2018. Transfer learning for named-entity recognition with neural networks. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Bill Yuchen Lin and Wei Lu. 2018. Neural Adaptation Layers for Cross-domain Named Entity Recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2012–2022, Brussels, Belgium. Association for Computational Linguistics.

Zihan Liu, Genta Indra Winata, Peng Xu, and Pascale Fung. 2020. Coach: A Coarse-to-Fine Approach for Cross-domain Slot Filling. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 19–25, Online. Association for Computational Linguistics.

Wentao Ma, Yiming Cui, Chenglei Si, Ting Liu, Shijin Wang, and Guoping Hu. 2020. CharBERT: Character-aware pre-trained language model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 39–50, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Thomas L. Packer, Joshua F. Lutes, Aaron P. Stewart, David W. Embley, Eric K. Ringger, Kevin D. Seppi, and Lee S. Jensen. 2010. Extracting person names from diverse and noisy ocr text. In *Proceedings of the Fourth Workshop on Analytics for Noisy Unstructured Text Data*, AND '10, page 19–26, New York, NY, USA. Association for Computing Machinery.

Danish Pruthi, Bhuwan Dhingra, and Zachary C. Lipton. 2019. Combating Adversarial Misspellings with Robust Word Recognition. *arXiv:1905.11268 [cs]*. ArXiv: 1905.11268.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*.

Afshin Rahimi, Yuan Li, and Trevor Cohn. 2019. Multilingual NER transfer for low-resource languages. *CoRR*, abs/1902.00193.

Danny Rodrigues Alves, Giovanni Colavizza, and Frédéric Kaplan. 2018. Deep reference mining from scholarly literature in the arts and humanities. *Frontiers in Research Metrics and Analytics*, 3:21.

Kepa J. Rodriguez, Mike Bryant, Tobias Blanke, and Magdalena Luszczynska. 2012. Comparison of named entity recognition tools for raw ocr text.

Devendra Singh Sachan, Pengtao Xie, Mrinmaya Sachan, and Eric P. Xing. 2018. Effective Use of Bidirectional Language Modeling for Transfer Learning in Biomedical Named Entity Recognition. *arXiv:1711.07908 [cs]*. ArXiv: 1711.07908.

Stefan Schweter and Luisa März. 2020. Triple E - effective ensembling of embeddings and language models for NER of historical german. 2696.

Lichao Sun, Kazuma Hashimoto, Wenpeng Yin, Akari Asai, Jia Li, Philip Yu, and Caiming Xiong. 2020. Adv-BERT: BERT is not robust on misspellings! Generating nature adversarial samples on BERT. *arXiv:2003.04985 [cs]*. ArXiv: 2003.04985.

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2006. ACE 2005 Multilingual Training Corpus. Type: dataset.

Jing Wang, Mayank Kulkarni, and Daniel Preotiuc-Pietro. 2020a. Multi-Domain Named Entity Recognition with Genre-Aware and Agnostic Inference. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8476–8488, Online. Association for Computational Linguistics.

Xinyu Wang, Yong Jiang, Nguyen Bach, Tao Wang, Zhongqiang Huang, Fei Huang, and Kewei Tu. 2020b. Automated concatenation of embeddings for structured prediction. *CoRR*, abs/2010.05006.

Xinyu Wang, Yong Jiang, Nguyen Bach, Tao Wang, Zhongqiang Huang, Fei Huang, and Kewei Tu. 2021. Improving named entity recognition by external context retrieving and cooperative learning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1800–1812, Online. Association for Computational Linguistics.

Zhenghui Wang, Yanru Qu, Liheng Chen, Jian Shen, Weinan Zhang, Shaodian Zhang, Yimei Gao, Gen Gu, Ken Chen, and Yong Yu. 2018a. Label-aware double transfer learning for cross-specialty medical named entity recognition. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1–15, New Orleans, Louisiana. Association for Computational Linguistics.

Zhenghui Wang, Yanru Qu, Liheng Chen, Jian Shen, Weinan Zhang, Shaodian Zhang, Yimei Gao, Gen Gu, Ken Chen, and Yong Yu. 2018b. Label-Aware Double Transfer Learning for Cross-Specialty Medical Named Entity Recognition. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1–15, New Orleans, Louisiana. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Jiateng Xie, Zhilin Yang, Graham Neubig, Noah A. Smith, and Jaime G. Carbonell. 2018. Neural cross-lingual named entity recognition with minimal resources. *CoRR*, abs/1808.09861.

Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. 2020. LUKE: Deep contextualized entity representations with entity-aware self-attention. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6442–6454, Online. Association for Computational Linguistics.

Huiyun Yang, Shujian Huang, Xin-Yu Dai, and Jiajun Chen. 2019. Fine-grained Knowledge Fusion for Sequence Labeling Domain Adaptation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4195–4204, Hong Kong, China. Association for Computational Linguistics.

Joey Tianyi Zhou, Hao Zhang, Di Jin, Hongyuan Zhu, Meng Fang, Rick Siow Mong Goh, and Kenneth Kwok. 2019. Dual Adversarial Neural Transfer for Low-Resource Named Entity Recognition. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3461–3471, Florence, Italy. Association for Computational Linguistics.

TFW2V: An Enhanced Document Similarity Method for the Morphologically Rich Finnish Language

Quan Duong,[1] **Mika Hämäläinen,**[1,2] **Khalid Alnajjar**[1,2]
firstname.lastname@{helsinki.fi}
[1]University of Helsinki, [2]Rootroo Ltd, Finland

Abstract

Measuring the semantic similarity of different texts has many important applications in Digital Humanities research such as information retrieval, document clustering and text summarization. The performance of different methods depends on the length of the text, the domain and the language. This study focuses on experimenting with some of the current approaches to Finnish, which is a morphologically rich language. At the same time, we propose a simple method, TFW2V, which shows high efficiency in handling both long text documents and limited amounts of data. Furthermore, we design an objective evaluation method which can be used as a framework for benchmarking text similarity approaches.

1 Introduction

Identifying documents that describe similar topics is a challenging yet important task. Detecting similar documents automatically has a wide range of digital humanities applications such as OCR post-correction (Dong and Smith, 2018), automatic clustering and linking of documents (Arnold and Tilton, 2018; Riedl et al., 2019) and clustering of semantic fields within a document (Hämäläinen and Alnajjar, 2019).

Assessing document similarity automatically becomes an important task especially due to the often unstructured nature of digital humanities research data (see Mäkelä et al. 2020). This makes it possible to handle large text corpora in a more organized fashion by clustering similar texts together.

In this paper, we explore different approaches to textual similarity detection, namely TF-IDF, USE, Doc2Vec and our own proposed approach named TFW2V[1]. Our approach combines the traditional TF-IDF method with word embeddings to improve the overall performance of the text similarity method. Unlike the recent neural approaches such as BERT (Devlin et al., 2018), RoBERTa (Liu et al., 2019) or XLNet (Yang et al., 2019b), our method does not rely on a large external corpus, but can be fully trained on the texts the similarity of which one is to assess. This is useful since our model can then work on corpora that represent a different era than what modern NLP models are trained on, or even for languages that do not have massive text collections readily available or are spoken by communities that do not have access to the computational resources needed to train large neural language models.

2 Related work

A survey conducted by Beel et al. (2016) showed that 83% of text-based recommendation systems in digital libraries use TF–IDF. There is also a recent survey paper on the current state of Finnish NLP (Hämäläinen and Alnajjar, 2021). There is a number of papers studying automatic detection of genres (Dalan and Sharoff, 2016; Worsham and Kalita, 2018; Gianitsos et al., 2019), which, as a task, is not too far from ours. However, in this section, we focus mainly on approaches on document similarity.

Kim et al. (2019) have combined multiple document representation approaches, which are TF-IDF, LDA and Doc2Vec, to classify documents in a semi-supervised fashion. Their results indicate that combining the features of the aforementioned models enhanced the performance of the classification task. Truşcă (2019) has compared how different text representation models perform when training a Support Vector Machine (SVM) classifier. The results show that Doc2Vec was the superior model for the task addressed by the author, which is text categorization. Duong et al. (2021b) also showed

[1]Code available: https://github.com/ruathudo/tfw2v

Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH), pages 163–172
December 19, 2021.

that clustering Finnish text is more effective by Doc2Vec compared to LDA.

A recent study by Marcińczuk et al. (2021) compared WordNet –a manually constructed network of concepts–, TF-IDF and word embeddings extracted from Doc2Vec and BERT for unsupervised classification of Polish text documents. Their study showed that manually constructed knowledge bases, i.e. WordNet in this case, is a valuable resource for the task. Yang et al. (2016) merged TF-IDF and Word Embeddings similarity scores to build the recommendation system for similar bug reports.

Li et al. (2019) have used text representation models to extract keywords from short texts collected from social media by employing a TextRank (Mihalcea and Tarau, 2004) algorithm which constructs a network and traverses it using random walk to discover the most important concepts. Text representation models have also been utilized with deep neural networks to classify text by Dogru et al. (2021). TF-IDF and word embeddings have also been used to assess the similarity of entities (Hämäläinen et al., 2021). In particular, the authors used the aforementioned methods to extract and predict properties for Pokémon.

3 Experiments

In this section, we apply four of the existing approaches to predict similarity of documents: Doc2Vec, Universal Sentence Encoder (USE), term frequency–inverse document frequency (TF-IDF) and average weighted word vector (AvgWV). Later on, we propose a new method to optimize TF-IDF by using a word embeddings model (TFW2V). All the experiments use the same datasets, the sampling process of which will be presented in the following section.

3.1 Dataset

We run the experiments based on the Yle News corpus. This corpus contains news articles published from 2011 to 2018 by the Finnish broadcasting company Yle (Yleisradio). The corpus is distributed through the Language Bank of Finland (Kielipankki)[2] and is freely available for research use[3]. There are more than 700,000 articles written in Finnish, each of which belongs to different categories with top-level categories such as Sport, Politics and Transportation. These categories have been defined by human authors and they have been coupled with keyword tags. For example, an article about a hockey match has the tags: *urheilu* (sports), *jääkiekon* (ice-hockey), *miesten* (men's), *sm-liiga* (The Finnish National League). The keyword tags illustrate well the contents of each article.

Our study focuses on tackling the text similarity problem for documents as opposed to individual sentences or paragraphs. For this reason, we decided to filter the corpus to include only the articles that are between 200 and 600 words for the experiments. Next, we randomly sample 10 datasets from the filtered corpus so that each dataset contains 2000 unique articles. Thus, all datasets are independent from each other with no overlap. We only optimize the models for the first dataset as the training set. For testing, the models are applied to the rest of datasets with the extracted parameters without any modification.

3.2 TF-IDF

The first method we experiment with is TF-IDF. As stated in section 2, this is a very simple method but it is very effective in many cases. The idea of this method can be expressed as follows: In a document, if a term (word) appears more frequently, it is given more weight, or a more important score. In contrast, if a term appears in many other documents in the corpus, it is regarded as less important or assumed to be a common word not descriptive enough for the document. The concurrence of these two metrics is combined in the equation below, to indicate the importance of a term in the text.

$$W_{i,j} = TF_{i,j} \times \log(\frac{N}{DF_i})$$

In this equation, the $W_{i,j}$ is the weight of a term i in document j, $TF_{i,j}$ is frequency of term i in document j, N is the number of documents in the corpus and DF_i refers to the number of documents where the term i appears. The weights hence tend to filter out common terms and emphasize the important keywords of a given document. The value of TF-IDF weight is in range $[0, 1]$.

Before running the experiment, the text data is cleaned by removing punctuation and stopwords using NLTK (Bird et al., 2009). For each sampled dataset, we calculate the TF-IDF weights for all documents. The pairs of terms and weights are feature vectors for each document. By using the

[2] http://urn.fi/urn:nbn:fi:lb-2017070501

[3] According to the license we cannot redistribute datasets derived from these data.

cosine similarity function, we can measure the similarity between feature vectors. In order to not depend on magnitudes of vectors but their angles, this is a common metric to compute the semantic similarity for encoded text (Singhal, 2001). After having similarity scores calculated, they are saved for each pair of documents in dataset and sorted in descending order. From now, the top N similar documents can be queried from a given document.

3.3 Average Weighted Word Vectors

Extending from the previous section 3.2, we introduce a combined method between TF-IDF and word embeddings algorithms called average weighted word vectors (AvgWV). This method was used in several previous researches to get a better representation for text document. Rani and Lobiyal (2021) used this method to get the representation of sentences in document to find the similarity between them. With the same approach, Charbonnier and Wartena (2018) applied to map the definition of an acronym with it context. The idea of this method is very easy to conduct. Both word embeddings and TF-IDF are trained for the given corpus. The representation of a document is the average of embedded vectors multiplied with the TF-IDF scores (weights) for all words in that document. By that, the TF-IDF scores punish the insignificant words and the influenced words have more impact on the averaged vector. The equation below is used to formulate the method.

$$\overrightarrow{D} = \frac{1}{N}\sum_{i}^{N} TF_i * \overrightarrow{WV}_i$$

Where $\overrightarrow{D}$ is the vector representation of a document, N is the number of word features. For each word i, we calculate the product of its TF-IDF score (TF) with its word vector ($\overrightarrow{WV}$) to get a new weighted vector. All weighted vectors corresponding to the word features are then averaged as the representation of document D.

The word embeddings model used in our experiment is based on Word2Vec from the work of Mikolov et al. (2013). The model was trained in 20 epochs using the Gensim library with a vector size of 128, skip-gram method, negative windows of 5 for each sample dataset. Inherit from previous TF-IDF section 3.2, the averaged vectors are applied cosine distance to get the similarity score for documents.

3.4 Doc2Vec

Documents originally stored in text format are convenient for humans to read, but they pose a challenge for computational tasks. Transforming from characters to a fixed length numeric representation is helpful for many purposes, for example: document retrieval, semantic comparison. One vector representation of text has been introduced by Harris (1954) as a bag of words method. Even though this method is easy to compute and it shows the efficiency in many cases, there are still some weaknesses such as the lack of importance given to the word order and it suffers from data sparsity and high dimensionality. It is also missing the semantic meaning between the words, for example, the words “dog” and “cat” are more similar than “dog” and “car” but they are treated equally in the Bag of Words method.

Doc2Vec (Le and Mikolov, 2014) is a document embeddings algorithm that comes to solve the issue from Bag of Words. The advantage of Doc2Vec is to vectorize a whole text document regardless of its length and to provide the semantic relationship of words.

We use the existing implementation of Doc2Vec in Gensim library (Řehůřek and Sojka, 2010). Before training, the text is tokenized and all stopwords are removed using NLTK. The setup on Doc2Vec model is kept in default with a dimensionality to 100 for vector size, negative sampling of 5 words and train for 30 epochs. We train model for each dataset separately. Thus there are 10 different Doc2Vec models corresponding to the datasets. Cosine similarity is again applied to these dense document vectors from the Doc2Vec model to get the similarity scores between documents.

3.5 Universal Sentence Encoder

The next approach we experienced is using the Universal Sentence Encoder (USE) (Yang et al., 2019a) model for multi-languages. The USE model was trained based on the Transformer architecture (Vaswani et al., 2017) for over 16 languages which shows a very good performance for various semantic textual similarity tasks. However, this model does not support Finnish.

Reimers and Gurevych (2020) introduced a novel way to transfer knowledge of a sentence encoder model from one language to another. On that paper, DistilmBERT (Sanh et al., 2019) model, a distilled version of BERT (Devlin et al.,

2018) trained on 104 different languages [4], was selected as student model. It is then adapted to USE model (Yang et al., 2019a) (as a teacher model) to support 50+ languages including Finnish. The pre-trained model was published with the name "distiluse-base-multilingual-cased-v2" in the Sentence-Transformers library (Reimers and Gurevych, 2019).

We applied the pre-trained model without any modifications. The maximum length support for the text is 512 tokens. The whole document is encoded automatically by the model and output as a dense vector. With the collected vectors we are able to compare the similarity between documents using cosine similarity.

3.6 Enrich TF-IDF by Word Embeddings (TFW2V)

The following part of this paper moves on to describe our modified version of TF-IDF algorithm. As introduced in section TF-IDF 3.2, this algorithm is very simple to compare similarity of documents. However, it also has many drawbacks. Firstly, the position of words in text is completely ignored. Secondly, because of relying on the lexical features, it skips semantic relationship of words. For example, with the synonyms or plural form of words, TF-IDF treats them as separated features without any linking. This will have a huge impact on morphologically rich languages such as Finnish, which contains many inflectional forms for all words and their compounds (see Duong et al. 2021a) even when lematization is applied. To overcome the issues, we propose a new algorithm that uses a word embeddings model to enrich the TF-IDF result. The details of the algorithm are presented in pseudo code 1.

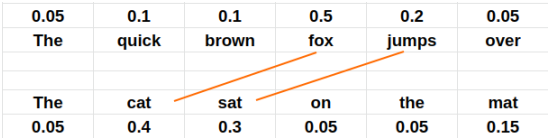

Figure 1: The two texts have the words **Fox** and **Cat** with high TF-IDF weights. At the same time, they have semantic similarity in the Word2Vec model, so that the documents can be linked. Same is applied to the words **Jumps** and **Sat**

The general idea of the algorithm can be explained as follows. We train a word embeddings model from the same corpus, so the words or terms of documents have semantic relationships. The word embeddings model can be used to measure the similarity of two terms. Turning now to TF-IDF output, the terms or features of a document contain the important information with higher weights. These important terms of two documents can be semantically linked by using a trained word embeddings model. An example is shown in the figure 1 to better explain.

The level of similarity between two group features is used to give additional reward on the final similarity score between a pair of documents. For example, if document A has important features (term1, term3, term8) and document B has important features (term2, term5, term9), the similarity score between document A and B can be added a small portion from the semantic similarity score between two features group. Similar to AvgWV section 3.3, the Word2Vec (W2V) model was used for word embeddings. The model was trained in 20 epochs using the Gensim library with a vector size of 128, skip-gram method, negative windows of 5 for each sample dataset.

To determine how much reward should be added to the TF-IDF similarity score, we design three parameters: **MinWeight**, **MaxTerm** and **Alpha**. Let take a look at the algorithm 1. Given a list of features (terms with weights) from a document and a list of similar documents as the result from TF-IDF, we want to change the result or re-rank it. Firstly, the features are sorted in the descending order of weight. The **MinWeight** parameter is used to filter important features, higher it is, less features are kept for comparison (lower bound). In some cases, the number of features considered as essential is too high, and we want to trim them to a certain number by the **MaxTerm** number (upper bound). For the list of similar documents, we apply the same process. After that, we get the similarity score between given features and compared features by W2V model. Note that, the W2V model generated by Gensim provides a method to compute similarity score of two set of words by averaging vectors for each set [5]. Next, the new similarity score is calculated by the following formula:

$$NewScore = \frac{WVScore \times Alpha + SimScore}{1 + Alpha}$$

[4] Provided through the Transformers Python library (Wolf et al., 2019) https://huggingface.co/distilbert-base-multilingual-cased

[5] https://radimrehurek.com/gensim/models/keyedvectors.html#gensim.models.keyedvectors.KeyedVectors.n_similarity

Algorithm 1 Enrich TF-IDF

procedure ENRICHTFIDF($Features, SimDocs, W2V, MinWeight, MaxTerm, Alpha$)
  sort $Features$(TermID, Weight) in DESC order of Weight
  filter $Features$ have $Weight < MinWeight$
  trim $Features$ to $MaxTerm$ **if** length $Features > MaxTerm$
  for $SimFeatures, SimScore$ in $SimDocs$ **do**
    sort $SimFeatures$(TermID, Weight) in DESC order of Weight
    filter $SimFeatures$ have $Weight < MinWeight$
    trim $SimFeatures$ to $MaxTerm$ **if** length $SimFeatures > MaxTerm$
    $WVScore = W2V.calculate\_similarity(Features, SimFeatures)$
    $NewScore = (WVScore \times Alpha + SimScore)/(1 + Alpha)$
    save $NewScore$
  end for
end procedure

Where the **SimScore** is the similarity score from TF-IDF, **WVScore** is the similarity score from W2V model for the important features, and **Alpha** is the parameter to decide how W2V similarity influences the current score. When **Alpha** is equal to 0, it has no effect, and when it is set to 1, the new score is the average of the two scores. Larger **Alpha** will have a higher recall which is bound to link an increasing number of unexpected documents together, while a smaller number yields a more conservative end result. In our experiments, we empirically set $MinWeight = 0.08$, $MaxLength = 20$ and $Alpha = 0.1$. These parameters are consistent for all datasets. Finally, the results are re-ranked for the new similarity scores.

4 Evaluation

Turning now to the evaluation, as previously stated, we have 10 independent datasets for benchmarking. All the experimented parameters from the models are applied consistently for those datasets. We assess the performance of the 5 methods TF-IDF, AvgWV, Doc2Vec, USE and TFW2V by three criteria: Top-N Precision, Top-N BLEU score and Top-N ranking loss. We will explain those metrics in the following sections of the paper.

4.1 Ground Truth

The ground truth for evaluation is created by the tags attached to articles. Because the tags are manually labeled by human authors to illustrate the content of articles, comparing the similarity between sets of tags can reflect the similarity of articles. There are many ways to measure similarity of two sets, such as counting overlapping tags. In machine translation, BLEU score (Papineni et al., 2002) is a popular method to evaluate the translated sentence quality. BLEU method calculates the similarity between two sets of words, very close to our case. The difference is only that two sentences in machine translation have N-grams dependence while similarity of two sets of tags are not relied on the position of tags. We use a simplified version of BLEU score, which is calculating score for unigrams without considering other higher order N-grams. After having BLEU scores for all document pairs, we sort them in descending order for evaluation.

4.2 Top-N Precision

The first metric is Top-N Precision. The metric can be presented as in the Top-N documents predicted as the most similar to a given one, i.e. how many documents are correctly ranked. For instance, given a document with top 100 similar documents, there are 40 documents that are ranked correctly in the top 100, the precision for it is 40%. The precision is calculated for all 2000 documents and averaged for each dataset. The formula below is to calculate the precision, where the D_{pred} is a set of predicted documents, D_{real} is a set of ground truth, and N is the number of documents for Top-N:

$$Precision = \frac{\sum D_{pred} \cap D_{real}}{N}$$

In the figure 2, the precision for Top-30 is presented. For all the 10 datasets in figure 2, the TFW2V model outperforms the other models clearly with an **26.09%** average (avg) accuracy. The Doc2Vec model has the lowest accuracy (**12.70%** avg). While the USE model shows

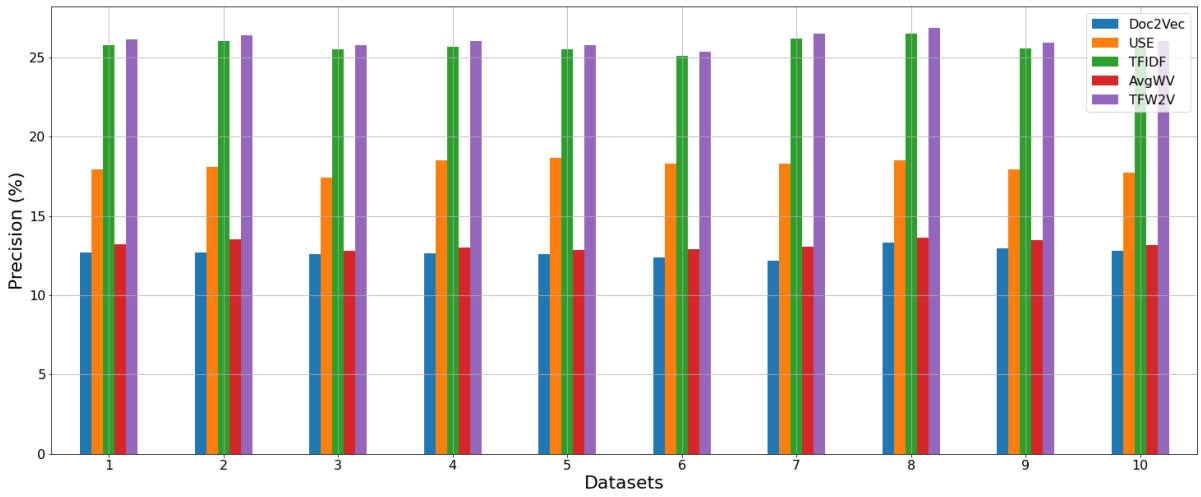

Figure 2: Precision accuracy for all datasets on Top-30 similarity. Higher is better.

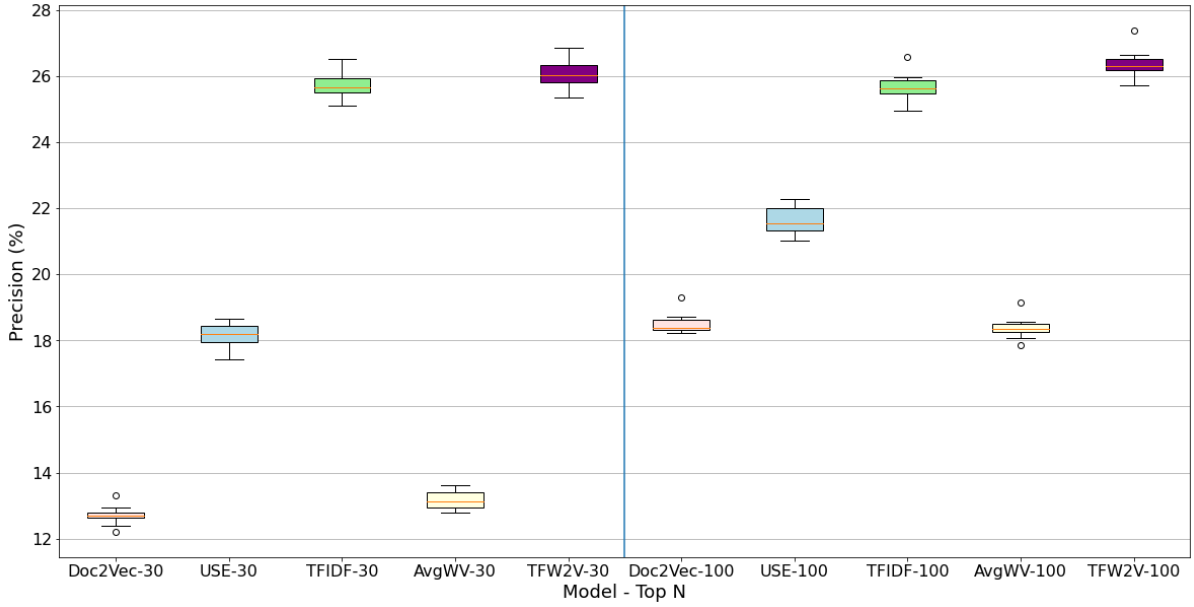

Figure 3: Precision accuracy for all datasets on Top-30 and Top-100 similarity. Higher is better.

much better results compared to Doc2Vec (**18.15%** avg), it is still inferior to TF-IDF (**25.76%** avg). The AvgWV is approximately comparable with Doc2Vec with slightly better numbers (**13.17%** avg). Similar results also take place for the Top-100, where the TFW2V surpassed the other models in every dataset. Therefore, to have a better visualization, we use boxplot to illustrate not only the difference between models, but also the Top-N variants.

The figure 3 demonstrates the precision for both Top-30 and Top-100 results. It is interesting that the results from Doc2Vec (**18.51%** avg), AvgWV (**18.38%** avg) and USE (**21.64%** avg) are significantly improved for the Top-100. This can be explained as the more relaxing the boundary, the higher the chance a document is predicted correctly in the Top-N list. However, the TF-IDF result has not improved. Despite depending on the TF-IDF results, TFW2V-100 still shows a slight increase (**26.41%** avg) compared to Top-30 results.

4.3 Top-N BLEU score

The next metric is Top-N BLEU score to measure how relevant a group of similar documents is to a given document. The way to conduct this metric is very similar to calculating the BLEU score for ranking in the Ground Truth section 4.1. We calculate the BLEU score on a unigram level for the tags of a given document against the Top-N similar

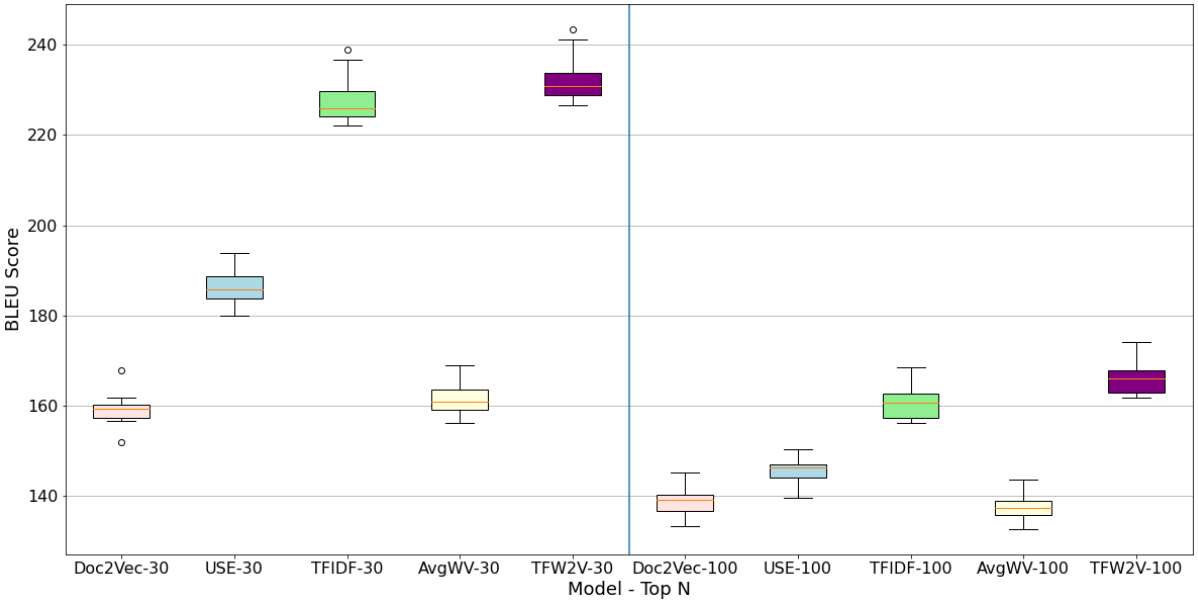

Figure 4: Sum of BLEU scores for 2k documents in each dataset. All datasets are evaluated for Top-30 and Top-100 similarity. Higher is better.

documents tags. These scores are then averaged for N similar documents. From there, we sum all the averaged BLEU scores for 2000 documents in a dataset. The averaged BLEU score is calculated as follow:

$$Score = \frac{\sum_{i=1}^{N} BLEU(T, T_i)}{N}$$

where T is the tags from a given document and T_i is the tags from similar documents and N is the number of Top-N. From the results we observed, the TFW2V model again outperforms the other models in all datasets. The boxplot in figure 4 shows the precise expression of performances for all models. We can see, Doc2Vec still remains less effective, around **160** for Top-30 and **139** for Top-100 on average. Similar numbers to Doc2Vec come from AvgWV method. The USE model results (**186** and **145**) are still lower compared to TF-IDF (**228** and **160**). Our proposed model TFW2V showed the improvement to TF-IDF with **233** and **166** scores on both Top-30 and Top-100 respectively.

The overall result for Top-30 is higher than Top-100. This is understandable as the more documents in Top-N there are, the more irrelevant ones making to the list will make the average scores decrease.

4.4 Top-N Ranking loss

The final metric we want to introduce is Top-N Ranking loss. This metric reflects how far a predicted position of similar documents is to the real order in ground truth for the Top-N. For example, in Top-30, we compare 30 predicted document orders to their real orders. If a predicted document has the position 5 and its real position is 45, the loss between the two orders is 40. Thus, the average loss for all documents is calculated using the Mean Absolute Error (MAE) function. The MAE loss is then divided for the length of the dataset (length of max rank) for normalization. The formula below is for calculating MAE loss between two positions P (ground truth) and $\hat{P}$ (prediction) for each document in Top-N with S is the length of the dataset.

$$Loss = \frac{\sum_{i=1}^{N} |P_i - \hat{P}_i|}{N \times S}$$

We got the result for this metric illustrated in figure 5. This time, both Doc2Vec and AvgWV models show the highest loss at Top-30 with loss around **0.29**. Interestingly, they are a bit better than the USE model at Top-100 (**0.30** vs **0.31**). TF-IDF is still obviously impressive compared to the previous ones with **0.24** for Top-30 and **0.29** for Top-100. Though, TFW2V is continuing to achieve the best result with the lowest losses of **0.23** and **0.28** for Top-30 and Top-100 respectively. This also indicates that the TFW2V model gives less irrelevant documents in Top-N than the other models.

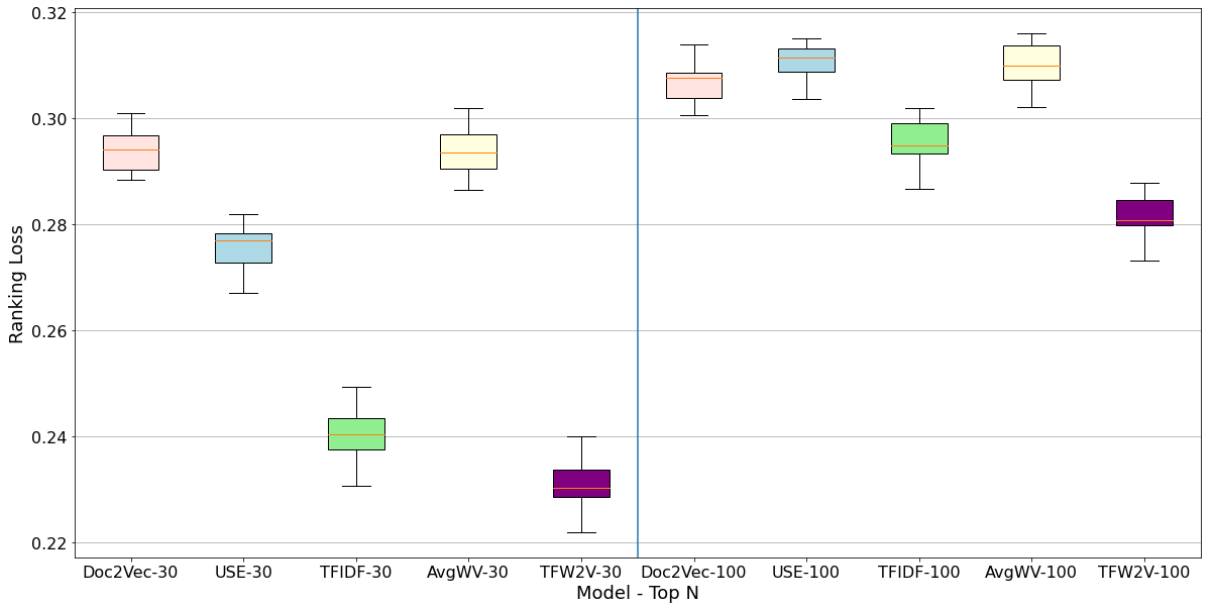

Figure 5: MAE Ranking loss for all datasets on Top-30 and Top-100 similarity. Lower is better.

5 Conclusion

In summary, we have presented a simple method to improve the TF-IDF algorithm by using a word embeddings model. The proposed method outperforms the more complex models like Doc2Vec and USE. We also compared it to a popular method AvgWV which use the same combination of TF-IDF and Word2Vec but in different way. It is very obvious that our proposed approach is surpassing the AvgWV model. The weakness of AvgWV is that it's hard to control the averaged vector representation of a document when all words and their TF-IDF weights are taken into account. Additionally, the impact from word vectors could come to too strong in some cases, which create the bias in the final decision.

In our method TFW2V, we can control the effect of word embedding model on the similarity score. At the same time, not all the words are considered into the enhancing process but the important ones. Thus, it is more stable, flexible and controllable to apply in various purposes. For example, in the document retriever system, the parameters can be set to get the relevant result as priority. On the other hand, in a recommender system, the parameters can be adjusted to get more creative result, thus it can look up for the under-discovered articles.

It is clearly observable that with a morphologically rich language like Finnish, TF-IDF still works very effectively. However, by combining it with a Word2Vec model and our algorithm 1, the result is significantly enhanced. The method is entirely unsupervised and works well with a small dataset like in our experiment with only 2000 samples.

In the future work, we will experiment this method for more languages and different lengths of document. The source code of this project will be provided as a Python library[6] which is easy to install and apply for any DH research. The lack of dependency on neural language models trained on massive amounts of data makes our approach applicable in scenarios where such amounts of text are unfeasible to obtain.

References

Taylor Arnold and Lauren Tilton. 2018. Cross-discourse and multilingual exploration of textual corpora with the DualNeighbors algorithm. In *Proceedings of the Second Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 50–59, Santa Fe, New Mexico. Association for Computational Linguistics.

Joeran Beel, Bela Gipp, Stefan Langer, and Corinna Breitinger. 2016. Research-paper recommender systems : a literature survey. *International Journal on Digital Libraries*, 17(4):305–338.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

[6] https://github.com/ruathudo/tfw2v

Jean Charbonnier and Christian Wartena. 2018. Using word embeddings for unsupervised acronym disambiguation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2610–2619, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Erika Dalan and Serge Sharoff. 2016. Genre classification for a corpus of academic webpages. In *Proceedings of the 10th Web as Corpus Workshop*, pages 90–98, Berlin. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of deep bidirectional transformers for language understanding.

Hasibe Busra Dogru, Sahra Tilki, Akhtar Jamil, and Alaa Ali Hameed. 2021. Deep learning-based classification of news texts using doc2vec model. In *2021 1st International Conference on Artificial Intelligence and Data Analytics (CAIDA)*, pages 91–96.

Rui Dong and David Smith. 2018. Multi-input attention for unsupervised OCR correction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2363–2372, Melbourne, Australia. Association for Computational Linguistics.

Quan Duong, Mika Hämäläinen, and Simon Hengchen. 2021a. An unsupervised method for OCR post-correction and spelling normalisation for Finnish. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 240–248, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.

Quan Duong, Lidia Pivovarova, and Elaine Zosa. 2021b. Benchmarks for unsupervised discourse change detection. In *Proceedings of the 6th International Workshop on Computational History*.

Efthimios Gianitsos, Thomas Bolt, Pramit Chaudhuri, and Joseph Dexter. 2019. Stylometric classification of ancient Greek literary texts by genre. In *Proceedings of the 3rd Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 52–60, Minneapolis, USA. Association for Computational Linguistics.

Mika Hämäläinen and Khalid Alnajjar. 2019. Let's face it: Finnish poetry generation with aesthetics and framing. In *12th International Conference on Natural Language Generation*, pages 290–300, United States. The Association for Computational Linguistics.

Mika Hämäläinen and Khalid Alnajjar. 2021. The current state of Finnish NLP. In *Proceedings of the Seventh International Workshop on Computational Linguistics of Uralic Languages*, pages 65–72.

Mika Hämäläinen, Khalid Alnajjar, and Niko Partanen. 2021. How cute is Pikachu? gathering and ranking Pokémon properties from data with Pokémon word embeddings. *arXiv preprint arXiv:2108.09546*.

Zellig S Harris. 1954. Distributional structure. *Word*, 10(2-3):146–162.

Donghwa Kim, Deokseong Seo, Suhyoun Cho, and Pilsung Kang. 2019. Multi-co-training for document classification using various document representations: Tf–idf, lda, and doc2vec. *Information Sciences*, 477:15–29.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *International conference on machine learning*, pages 1188–1196.

Jun Li, Guimin Huang, Chunli Fan, Zhenglin Sun, and Hongtao Zhu. 2019. Key word extraction for short text via word2vec, doc2vec, and textrank. *Turkish Journal of Electrical Engineering & Computer Sciences*, 27(3):1794–1805.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Eetu Mäkelä, Krista Lagus, Leo Lahti, Tanja Säily, Mikko Tolonen, Mika Hämäläinen, Samuli Kaislaniemi, and Terttu Nevalainen. 2020. Wrangling with non-standard data. In *Proceedings of the Digital Humanities in the Nordic Countries 5th Conference*, number 2612 in CEUR Workshop Proceedings, pages 81–96, Germany. CEUR-WS.org.

Michał Marcińczuk, Mateusz Gniewkowski, Tomasz Walkowiak, and Marcin Będkowski. 2021. Text document clustering: Wordnet vs. TF-IDF vs. word embeddings. In *Proceedings of the 11th Global Wordnet Conference*, pages 207–214, University of South Africa (UNISA). Global Wordnet Association.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 404–411.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Ruby Rani and Daya K. Lobiyal. 2021. A weighted word embedding based approach for extractive text summarization. *Expert Systems with Applications*, 186:115867.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA. http://is.muni.cz/publication/884893/en.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. *arXiv preprint arXiv:2004.09813*.

Martin Riedl, Daniela Betz, and Sebastian Padó. 2019. Clustering-based article identification in historical newspapers. In *Proceedings of the 3rd Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 12–17, Minneapolis, USA. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108.

Amit Singhal. 2001. Modern information retrieval: A brief overview. *IEEE Data Eng. Bull.*, 24:35–43.

Maria Mihaela Truşcă. 2019. Efficiency of svm classifier with word2vec and doc2vec models. In *Proceedings of the International Conference on Applied Statistics*, volume 1, pages 496–503.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Joseph Worsham and Jugal Kalita. 2018. Genre identification and the compositional effect of genre in literature. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1963–1973, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Xinli Yang, David Lo, Xin Xia, Lingfeng Bao, and Jianling Sun. 2016. Combining word embedding with information retrieval to recommend similar bug reports. In *2016 IEEE 27th International Symposium on Software Reliability Engineering (ISSRE)*, pages 127–137.

Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernandez Abrego, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2019a. Multilingual universal sentence encoder for semantic retrieval.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019b. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.

Did You Enjoy the Last Supper? An Experimental Study on Cross-Domain NER Models for the Art Domain

Alejandro Sierra-Múnera
Hasso Plattner Institute
University of Potsdam
Potsdam, Germany
alejandro.sierra@hpi.de

Ralf Krestel
Hasso Plattner Institute
University of Potsdam
Potsdam, Germany
ralf.krestel@hpi.de

Abstract

Named entity recognition (NER) is an important task that constitutes the basis for multiple downstream natural language processing tasks. Traditional machine learning approaches for NER rely on annotated corpora. However, these are only largely available for standard domains, e.g., news articles. Domain-specific NER often lacks annotated training data and therefore two options are of interest: expensive manual annotations or transfer learning.

In this paper, we study a selection of cross-domain NER models and evaluate them for use in the art domain, particularly for recognizing artwork titles in digitized art-historic documents. For the evaluation of the models, we employ a variety of source domain datasets and analyze how each source domain dataset impacts the performance of the different models for our target domain. Additionally, we analyze the impact of the source domain's entity types, looking for a better understanding of how the transfer learning models adapt different source entity types into our target entity types.

1 Introduction

Cultural heritage archives contain vast amounts of unstructured data where valuable knowledge resides. This data can be analyzed and valuable information can be extracted using natural language processing (NLP) tools. Nowadays, most of the NLP tasks are performed using deep learning models which rely on large amounts of training data.

One of the core NLP tasks is named entity recognition (NER) which consists of finding mentions of *named entities* from a usually pre-defined set of entity types. Machine learning models learn entity and context patterns from labeled corpora allowing them to discover new entity mentions from unseen text.

In a scenario where there is a previously annotated large corpus, these models achieve good performance and can find new named entities from the pre-defined set of entity types. In the past, such datasets have been built for domains such as news wire (Tjong Kim Sang and De Meulder, 2003) and biomedical texts (Stubbs and Uzuner, 2015) containing annotations for entity types, such as *person*, *location*, *organization*, or *protein*, and *gene expression*.

Large labeled corpora are expensive and time-consuming to obtain. For less popular domains, large annotated corpora typically don't exist. There is especially a lack of annotated data for domain-specific entity types. One specific domain that requires non-standard entity types to be extracted is the cultural heritage domain. In this paper, we focus on digitized art-historic archives, in particular on the entity type *artwork*. For this particular entity type, there are no extensive datasets. The entity type *artwork* is different from the standard ones (person, location, organization, date) and poses some interesting challenges (Jain and Krestel, 2019). Not only is the entity type different, but also the structure and noise of digitized art-historic texts are different from news wire or biomedical text collections.

One particular challenge is the ambiguity inherent to the definition of such titles due to the fact that sometimes these titles describe a scene or contain other named entities. For instance, the painting titled '*Girl before a mirror*' by Pablo Picasso, depicts a girl before a mirror. Only the context of the mention identifies this phrase as a painting title.

Moreover, a big percentage of art-historic archives need to be digitized first using optical character recognition (OCR) software. This routinely introduces errors such as mis-identified characters, the addition of noise, and the loss of formatting structure(van Strien et al., 2020; Lin, 2003; Ro-

Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH), pages 173–182
December 19, 2021.

driquez et al., 2012). Further, the quality of OCRed texts strongly depends on the print quality of the original documents (Traub et al., 2015; Rodriquez et al., 2012; Mieskes and Schmunk, 2019).

Given the aforementioned challenges, different alternatives for solving the task could be explored. One would be manually annotating a large corpus with artwork title information. But besides being a time-consuming and expensive task, it would not scale to further cultural heritage entities such as galleries, art styles, or art movements. Another would be focusing on gazetteers and rule-based approaches. But, listing all possible artwork titles would not only be cumbersome, but would also not solve the ambiguity problem for phrases such as '*Girl before a mirror*'.

The most promising approach is to make use of existing, previously annotated corpora from other domains and *transfer* the learned patterns to the new domain. In combination with deep learning models, this domain adaptation via *transfer learning* or *multi-task learning* has shown good results for popular domains (Rodriguez et al., 2018). Different models have been proposed in the past under the concept of *cross-domain NER* to solve the problem. These models learn to identify named entities within a *target domain* based on patterns learned from a large, labeled dataset from a *source domain*.

The goal of this paper is to evaluate the performance of some of the best of those models for the artwork recognition task. The paper is structured as follows: In Section 2 we describe the existing cross-domain NER models. In Section 3 we describe the existing NER datasets available for different domains and the construction of a target dataset used for the training and evaluation of artwork recognition. In Section 4 we describe the evaluation setup and in Section 5 the results are outlined and finally, in Section 6, the conclusion and future work are proposed.

2 Related Work

In this section, we discuss previously proposed cross-domain NER models and focus on the domain adaptations that those models propose.

Cross-domain NER models could be divided into two main categories. The models in which the source and target domain share the entity types but have differences in terms of the vocabulary, and the models which consider the disparity between the entity types in the source and the target domain.

For instance, one traditional task for the first group would be the transfer of *persons*, *locations* and *organization* from the news domain into the social media domain. In this case, the persons mentioned in the news domain might be different from the persons mentioned in social media. Moreover, the language used in social media is different from the language used in news articles. Artifacts, such as emojis, hashtags, or '@' as well as the structure of sentences differ between domains. However, the entity types remain constant in the domain adaptation task. Liu et al. (2020a) propose a model for low-resource target domains combining multi-task learning (MTL) and a mixture of entity experts (MoEE), aiming to improve generalization and reduce the over-fitting effect when a model learns entities from a source domain. Zhou et al. (2019) propose a general neural transfer framework called Dual Adversarial Transfer Network (DATNet), Wang et al. (2020) extend the popular Bi-LSTM-CRF architecture for multi-domain NER, dividing the domain-specific and independent components of the network, to achieve adaptation over multiple genres.

Other models deal with different entity types in the target domain compared with the source domain. This is the case for domain-specific entity types where extensively annotated corpora are missing. Artwork mentions, for instance, are not annotated in traditional NER datasets, therefore we focus our study on this kind of domain adaptation. Within these models, Lee et al. (2018) proposed to transfer the weights of a Bi-LSTM-CRF model with both word and character embeddings. The weights were trained on the source domain and then fine-tuned on the smaller target dataset. They experimented with transferring different parts of the network to the target domain and concluded that transferring the weights from the lower layers of the network, particularly the character Bi-LSTM layer, improved the performance of the NER model on the target domain compared to a model trained only with the target dataset (no transfer). A similar model proposed by Lin and Lu (2018), called CDMA-NER augmented the idea of using a pre-trained model by including adaptation layers on top of it to perform the domain adaptation without the need of retraining the source model. The adaptation between domains is based on the bottom layer of the Bi-LSTM model, particularly on the adaptation of word embeddings.

A different kind of domain adaptation is used by models which simultaneously train the source and the target domain in a multi-task learning approach (Bhatia et al., 2018), (Beryozkin et al., 2019), (Jia and Zhang, 2020). In the multi-task model proposed by Jia and Zhang (2020) (Multi-Cell Compositional LSTM for NER Domain Adaptation) which is based on an LSTM network, each entity type has an independent cell state. Additionally a compositional cell combines all the entity type cells into the final output which is then passed to the domain-specific conditional random field (CRF). The domain adaptation is performed on the entity type level, and the model leverages the context embeddings provided by BERT (Devlin et al., 2019). In their experiments, they transferred information from the news domain to the biomedical and the social media domain.

Another recently proposed model called CrossNER (Liu et al., 2021) introduces domain-adaptive pre-training (DAPT) as a technique to continue the pre-training of language models such as BERT with domain-specific raw texts by masking spans of tokens instead of random tokens for training. They experimented with different masking strategies as well as different corpora selection criteria and concluded that the best performance is obtained when DAPT is performed in a set of sentences containing general and task specific entities. The entities used for corpora selection are chosen from predefined resources, such as gazetteers or knowledge graphs. Besides the pre-trained language model, which is trained in the target domain, the model uses a linear layer on top. They experimented with training the whole model on only the target domain, jointly training the source and target domain, and pre-traning in the source domain followed by fine-tuning on the target domain. Their results show that pre-training followed by fine-tuning yields better results. In their paper, they also introduce a new dataset with a diverse set of annotated texts from different domains with domain-specific entity types. In their experiments these domains are treated as targets. We use their dataset for our experiments but instead of treating them as target domains, we consider them as source domains. The details of the datasets we use are described in Section 3.

3 Datasets

As mentioned in Section 2, cross-domain NER relies on the knowledge of a source domain. This knowledge can be in the form of an annotated corpus with domain-specific entity types or a pre-trained model specialized in recognizing them. We use a diverse set of source datasets for this purpose. Regarding the target domain, we created a dataset with sentences containing art-related entity mentions. In the following subsections, we describe the datasets considered in our experiments as source datasets, as well as the target domain dataset which will serve as a training, validation, and test dataset.

3.1 Source Datasets

For source domain datasets, we consider the widely used CoNLL03 (Tjong Kim Sang and De Meulder, 2003) English dataset which consists of news texts annotated with the traditional named entity types: *person*, *location* and *organization* plus *miscellaneous*.

To study the impact of the source domain and its entity types, we also consider the dataset published by Liu et al. (2021): a collection of manually annotated corpora from five domains (artificial intelligence, music, literature, politics and science) that was labeled with domain-specific entity types. The variety in entity types is important in our evaluation because we focus on domain adaptation approaches that specifically need to deal with different entity types. In their paper, (Liu et al., 2021) used the newly labeled corpora as target domains, and the goal was to perform domain adaptation from the news domain to these, therefore the target training set was smaller than the validation and the test set, thus limiting the amount of labeled data in the target domain. In our experiment we consider those datasets as source datasets, therefore we split the corpora in a different way to increase the size of the training set.

Also these datasets contain only sentences mentioning at least one entity. Therefore, to have a fair comparison between source datasets, we filter the CoNLL03 dataset to keep only the sentences that mention an entity, we refer to the filtered dataset as the news dataset. This comprises the following reduction of sentences for the news dataset: the training set is reduced from 14,041 to 11,132 sentences, and the validation set is reduced from 3,250 to 2,605 sentences.

The resulting group of source datasets is referred to in our experiments as the unbalanced source datasets, due to the difference in sizes.

Additionally, in order to compare the impact of

Dataset	Balanced		Unbalanced	
	Train	Val.	Train	Val.
News	781	100	11132	2605
AI	781	100	781	100
Literature	781	100	816	100
Music	781	100	845	100
Politics	781	100	1192	200
Science	781	100	993	200

Table 1: Number of Sentences in Source Domain Datasets

	Train	Val.	Test
Sentences	180	70	294
Mentions	51	21	74

Table 2: Art Target Domain Dataset

the source domains and avoiding the possible bias of the dataset sizes, we under-sample each of the datasets except the AI dataset, being the smallest, to generate source datasets with exactly the same number of sentences in the training and validation sets. The resulting sizes in terms of sentences in the training and validation sets are detailed in Table 1.

3.2 Target Dataset

Our study focuses on the detection of artwork mentions in digitized art-historic documents. However, there is no public dataset available with artwork title annotations. Therefore, we manually annotated a set of randomly extracted 544 sentences for the evaluation of the different models. An annotation tool was used by two non-expert annotators, and afterwards the inter-annotator agreement in the results was analyzed. The Fleis-kappa (Fleiss, 1971) value was -1.86 and Krippendorf-alpha (Krippendorff, 1970) 0.61 meaning that there was poor agreement among annotators. As expected, even for humans, the annotation process was difficult due to the challenges expressed in Section 1. Therefore, an additional step of manual revision of each annotation was performed, and the disagreements were resolved with the help of web search. Afterwards, the target domain dataset was split into train, validation and test, with the sizes depicted in Table 2.

4 Experimental Setup

Our evaluation aims to shed light on the power of different cross-domain NER models to adapt to the art domain and recognize artwork mentions. For our experiments, we focus on the models **CDMA-NER** proposed by Lin and Lu (2018), **Muli-Cell LSTM** proposed by (Jia and Zhang, 2020) and **CrossNER** proposed by Liu et al. (2020b). To compare the performance, we train each of these models using datasets from a set of source domains and a single target domain training dataset. We measure the F1 score for the task of recognizing artwork mentions in the target test set.

For the experiments with CDMA-NER, GloVe (Pennington et al., 2014) word embeddings are used, and for both, Multi-Cell LSTM and CrossNER, which are designed to use pre-trained language models, we use BERT base model (cased) as well as an adaptation to the art domain following the domain-adaptive pre-training used in CrossNER.

4.1 Domain-Adaptive Pre-Training

We further pre-train the BERT base model (cased) for Multi-Cell LSTM and CrossNER with a set of raw art-related texts, we generated a set of 500,000 sentences extracted from digitized art-historic documents containing artwork titles from the Getty vocabularies (Harpring, 2010). Specifically, we perform a string match of sentences against the Cultural Objects Named Authority (CONA) vocabulary[1] and the Union List of Artist Names (ULAN)[2] containing titles of artwork and architecture, and artist names, respectively. With the 500,000 sentences, pre-training is performed for 15 epochs as proposed by Liu et al. (2020b).

4.2 Training

Each model was trained for a maximum of 500 epochs with early stopping and the validation set was used to determine when the model did not need further training and the best model was evaluated against the target test dataset. In the case of CDMA-NER and CrossNER, the source validation dataset was used to determine the best source model, before transferring the weights to the target domain

[1]Getty CONA (2017), http://www.getty.edu/research/tools/vocabularies/cona, accessed October 2021.

[2]Getty ULAN (2017), http://www.getty.edu/research/tools/vocabularies/ulan, accessed October 2021.

News	[Liam Gallagher:person] , singer of [Britain:location] 's top rock group [Oasis:organization] , flew out on Thursday to join the band three days after the start of its [U.S.:location] tour
AI	Examples of [supervised learning:field] are [Naive Bayes classifier:algorithm] , [Support vector machine:algorithm] , [mixtures of Gaussians:algorithm] , and network
Literature	It tied with [Roger Zelazny:writer] ' s [This Immortal:book] for the [Hugo Award:award] in 1966
Music	Two of his most popular recordings were [Layla:song] , recorded with [Derek and the Dominos:band] ; and [Robert Johnson:musical artist] ' s [Cross Road Blues:song] , recorded with [Cream:band]
Politics	Three [United States:country] presidents have been impeached by the [House of Representatives:misc] : [Andrew Johnson:politician] in 1868 , [Bill Clinton:politician] in 1998 , and [Donald Trump:politician] in 2019 .
Science	The journal establishment was similar to the starting of [The Astrophysical Journal:academic journal] and [The Astronomical Journal:academic journal] by [George Ellery Hale:scientist]
Art	Figure 39 . [On the Terrace:artwork] , 1867 . Panel , 17.7 x 18 cm . © The Cleveland Museum of Art , Bequest of Clara Louise Gehring Bickford , 1986.68 . Photo : Courtesy of the Museum .

Table 3: Dataset Examples

training. For all models their publicly available implementations were adapted to use the dataset configuration proposed in this paper.

Additionally, a baseline model was trained without using source domain data. This baseline model is based on the Bi-LSTM-CRF model originally proposed by Lample et al. (2016), and implemented using FlairNLP (Akbik et al., 2019). It was trained ten times using BERT base (cased) as the embedding model, the average F1 over the 10 runs is reported in Table 4.

The under-sampling process to generate the size-balanced source datasets is repeated 5 times to generate random subsets of the data. For the smaller AI source dataset, 5 shuffled versions with the same sentences are used to train the models. The results for the size-balanced experiments in Table 4 show the average performance over the 5 runs.

5 Results

Table 4 shows an overview of the results in terms of F1-measure for each of the evaluated models using the different source datasets, plus the performance of the baseline model. The first observation is that the baseline model achieves very competitive results in comparison to the cross-domain models. In only one occasion the other models were able to outperform the baseline, which suggests that the transfer learning approach seems to work in very specific settings.

Another observation is that DAPT is in general not improving the language model for the CrossNER model, which performs better with the original BERT model. One possible reason is the digitization noise introduced into the raw text used to perform DAPT. For Multi-cell LSTM, the average improvement is very small. The CDMA-NER model in general performs worse than the other models and the baseline, and the reason could be the lack of contextualized representation of words in the GloVe embeddings.

Generally, the CrossNER model performs better than the other two models and its performance is similar to the baseline, although the model is relatively simple in comparison to Multi-Cell. This suggests that the traditional LSTM-CRF combination might not be suitable for transfer learning to complex entities such as artworks. The combination of LSTM and CRF is positive for NER as shown by the performance of the baseline model, but as the architecture becomes more complex, the performance is compromised. Another reason why Multi-

Baseline: FlairNLP$_{BERT}$				.589			
Cross-Domain NER Models	**Source Domain**						
	News	**AI**	**Lit**	**Mus**	**Pol**	**Sci**	**Avg**
CDMA-NER	.460	.368	.344	.394	.409	.413	.386
Multi-cell LSTM$_{BERT}$	.255	.509	.385	.467	.438	.459	.451
Multi-cell LSTM$_{DAPT}$	.343	.487	.436	.464	.471	.413	.454
CrossNER$_{BERT}$	.537	.519	.578	.535	.521	.512	.533
CrossNER$_{DAPT}$	**.594**	.488	.507	.477	.482	.528	.496
	Size-balanced experiments						
CDMA-NER	.332	.339	.365	.336	.360	.318	.344
Multi-cell LSTM$_{BERT}$	.495	.455	.460	.446	.484	.441	.457
Multi-cell LSTM$_{DAPT}$	.434	.454	.489	.458	.415	.463	.456
CrossNER$_{BERT}$	.522	.535	.518	.543	.517	.586	.540
CrossNER$_{DAPT}$	.475	.503	.516	.560	.528	.518	.525

The results in bold font correspond to values higher than the baseline

Table 4: F1-Scores for Art Target Domain

Cell LSTM models might be performing worse than CrossNER is the fact that there is no overlap between source and target entity types, therefore the weights within the LSTM cells are not being strongly shared among domains.

The results of training the models with the unbalanced datasets reveal that the size of the source dataset does not guarantee a good target performance. The adapted news dataset is 13 times bigger than the music and literature datasets, but the performance is comparable when training the CrossNER$_{BERT}$ model. One reason for this behavior is the more general definition for entity types in CoNLL03, different from the more specialized entity types in the music and literature datasets.

One of the aspects which differentiate the various domains is the set of entity types that are relevant for the domain and are present in the different datasets. To study the impact on the performance of artwork recognition we remove individual entity types from the full music dataset. For each of the 13 entity types in this dataset, we generate an alternative version of the dataset in which the entity type is not considered in the annotations. This means that the tokens which were previously labeled as part of those named entities will remain in the dataset but without the annotation. Each altered dataset is used to train the 5 studied models. In Figure 1, the models' performance after altering the dataset is displayed as relative performance change with respect to the original experiment with the complete dataset. This way, we intend to analyze how each model depends on the source entity types to be able to transfer that knowledge to the recognition of artwork mentions.

From the figure it is clear that the Multi-cell LSTM model suffers a greater decrease in performance when the musical artists and bands are not present in the source dataset. This is an indicator of the manner in which this model learns the connections between the source and target entity types through the entity-typed LSTM cells. It is interesting, however, that in some cases the performance improves when removing entity types. This suggests that the model is sensitive to the similarity between the source and target entity types. Thus, depending on the type of entities we would like to recognize in the target domain, we should select the source dataset. Best results are achieved with the most similar entity types in the source domains. To phrase it in terms of the artwork recognition task, it would make sense to first analyze which domains contain titles of human-created creative works and then use those entity types exclusively.

Figure 2 depicts results of a similar experiment. In this case only one of the entity types is present in the dataset. Comparing both figures, it is clear that source datasets with just one entity type perform worse than source datasets with more variety in entity types. It is, however, counter-intuitive that the entity types which help the most in the transfer setting towards recognizing artworks are not

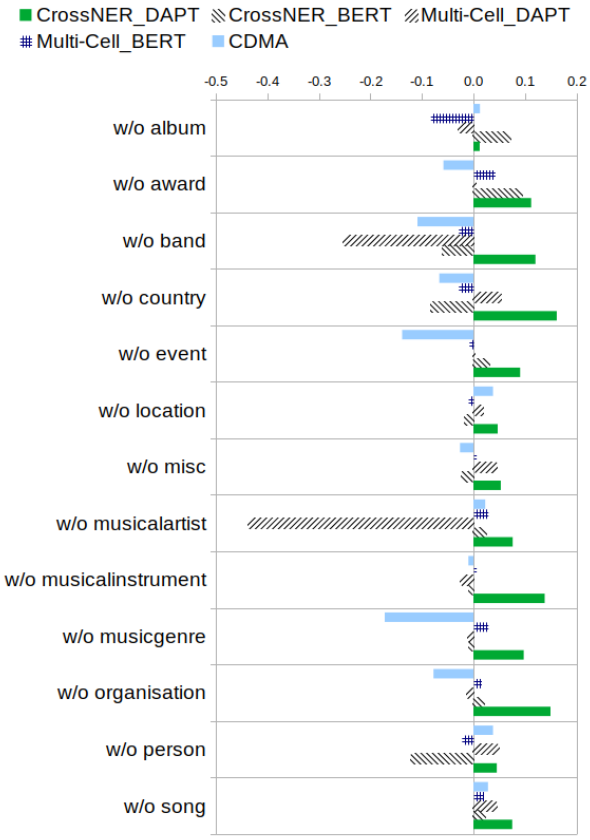

Figure 1: Change in F1 score when one entity type is removed from the source dataset *music*

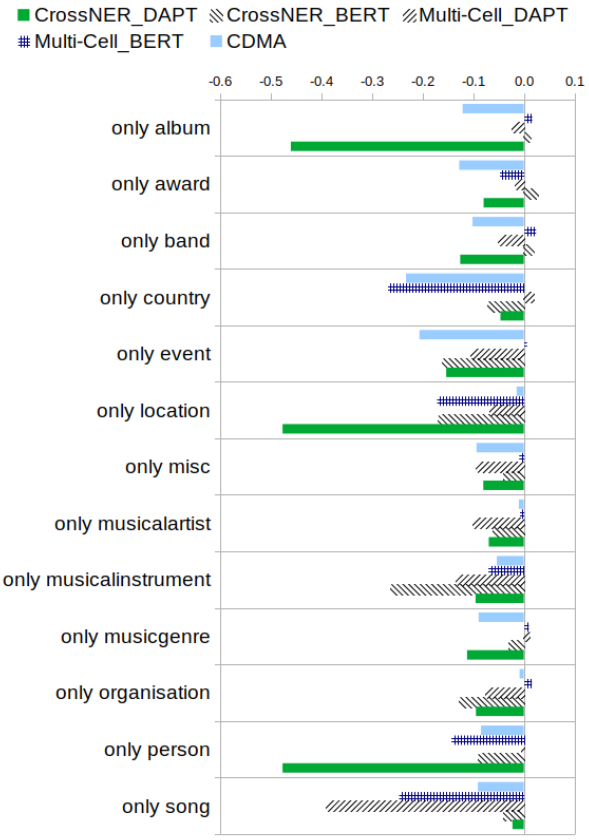

Figure 2: Change in F1 score when only using one entity type in the source dataset *music*

song or *album*, which are the entity types in the music domain that resemble closest to the notion of artwork.

Additional details of the results can be found in https://github.com/HPI-Information-Systems/cross-domain-ner

5.1 Qualitative Analysis

Besides the quantitative evaluation, we also performed an error analysis by investigating example predictions of the models. Specifically, we analyse the models trained with the original *music* source dataset.

Firstly, Table 5 example E1 shows a sentence which contains a correctly recognized entity mention and a typical error in which the model is able to recognize the presence of an artwork but the boundaries are not correctly identified. For other models the determiner of the second mention was part of the title, which is not the case in this particular example E1, but it is a persistent error for all models. The presence of the article in the titles is a complex boundary to define even for humans since there is no clear rule that could be applied.

In example E2, we see a false positive predicted by CDMA-NER and not predicted by any other model. One possible reason for this error is the lack of context in the sentence, the presence of a name at the beginning, and the quotation marks. Without the knowledge that Claude Monet is a painter, it would be hard to distinguish it from an artwork mention, given that many paintings are named after persons.

In example E3, the sentence is particularly long and contains many artwork mentions. The CrossNER$_{BERT}$ model, which is the best performing one, is able to identify all the mentions but fails to set the correct initial boundaries for three. One specially interesting observation is that 2 titles follow the pattern *'Painter* {*WORD*} *His* {*WORD*}*'* but the model is able to correctly recognize only one of them.

The fourth example E4 exemplifies a very challenging artwork title to recognize. It is a notably long title containing a combination of uppercase and lower case words and references to different locations. In our experiments, no model was able to recognize the artwork mention in that sentence.

E1	... as in [The Confidence] (Salon of 1857 , Fig . 26) . Even relaxed in a tavern , as in the [Smoker in Black] (Fig . 27) , the ... (Predicted by Multi-Cell LSTM$_{DAPT}$)
E2	19Robinson , " Claude Monet , " 698 (Predicted by CDMA-NER)
E3	The autobiographical dimension is furthered by Meissonier 's inclusion of works he had created or owned . [Painter Showing His Drawings] , set in the quai Bourbon studio,3 includes an enlarged [Samson Battling the Philistines] , perhaps hinting that the artist is similarly an inspired hero , coping with his own philistine world . Identifiable below is the unframed [Smoker] of 1842 ; in the center , [The Evangelists] , which is propped against a portrait recognizably of Meissonier ; and in the portfolio , a drawing for [The Evangelists] (Musée du Louvre , rf 1908) . In the background of [Painting Collectors] is also an enlarged version of the [Martyrdom of Saint Lawrence] used in the [Painter in His Studio] of 1843 and an Italian - school painting of a half - length , seminude woman that belonged to Meissonier.4 (Predicted by CrossNER$_{BERT}$)
E4	We are grateful to Cecilia Powell for pointing out the conflation in Wilton , op . cit . , of this watercolor with the [View down the Mosel from the Hillside above Pallien] (circa 1839 , illustrated in Powell , op . cit . , p. 132) , and to Peter Bower for his assistance in preparing this catalogue entry . (All models failed to recognize the mention)
E5	... lent from the distinguished collection of Mrs Walter Jones , the widow of Walter H. Jones . Her other loans included the [Red Rigi] (no . 891) , the [Blue Rigi] (no . 895) , [Venice , Mouth of the Grand Canal] (no . 899) and [Mainz and Castel] (no . 904) . When the drawing was sold ... (Predicted by CDMA-NER)

Squared brackets represent ground truth and highlighted text represents predicted annotations

Table 5: Inference Examples

6 Conclusions

In this paper, we studied the task of complex NER, specifically recognizing artworks in art-historic texts. We discuss the reasons why this is a hard task and why it is promising to leverage annotations from other domains to compensate for the lack of annotated resources for the art domain. We explained the concept of cross-domain NER using transfer learning which has been investigated in the past to achieve the aforementioned domain adaptation and presented related work connected to this concept. Based on the problem setup and a collection of annotated datasets, we performed a set of experiments to understand the performance of domain-adapted NER to recognize artworks. In the experiments we analyzed both, the models and the datasets, in order to isolate and understand independently different aspects of the presented approaches. From the experimental evaluation of Cross-domain NER approaches for the recognition of artworks we conclude that, although domain adaptation is a promising approach to achieve this goal, a simpler alternative, namely a LSTM-CRF model with BERT base (cased), perform as well as the best Cross-domain NER.

As future work, we would like to investigate the explainability and interpretability of cross-domain NER models to understand better their limitations and propose new models that not only take into account the differences in terms of entity types and language between domains, but also semantic relations between the domains and the named entities. Additionally, it would be of interest to investigate the domain adaptation of other tasks like information extraction and knowledge graph embedding models, which could be jointly trained with NER.

Acknowledgements

We thank the Wildenstein Plattner Institute for providing the digitized corpus used in this work. This research was funded by the HPI Research School on Data Science and Engineering.

References

Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. 2019. FLAIR: An easy-to-use framework for state-of-the-art NLP. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages

54–59, Minneapolis, Minnesota. Association for Computational Linguistics.

Genady Beryozkin, Yoel Drori, Oren Gilon, Tzvika Hartman, and Idan Szpektor. 2019. A joint named-entity recognizer for heterogeneous tag-sets using a tag hierarchy. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 140–150, Florence, Italy. Association for Computational Linguistics.

Parminder Bhatia, Kristjan Arumae, and Busra Celikkaya. 2018. Dynamic transfer learning for named entity recognition. *Studies in Computational Intelligence*, 843:69–81.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Joseph L. Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76:378–382.

Patricia Harpring. 2010. Development of the Getty vocabularies: AAT, TGN, ULAN, and CONA. *Art Documentation: Journal of the Art Libraries Society of North America*, 29(1):67–72.

Nitisha Jain and Ralf Krestel. 2019. Who is mona l.? identifying mentions of artworks in historical archives. In *Digital Libraries for Open Knowledge - 23rd International Conference on Theory and Practice of Digital Libraries, TPDL 2019, Oslo, Norway, September 9-12, Proceedings*, volume 11799 of *Lecture Notes in Computer Science*, pages 115–122. Springer.

Chen Jia and Yue Zhang. 2020. Multi-cell compositional LSTM for NER domain adaptation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5906–5917, Online. Association for Computational Linguistics.

Klaus Krippendorff. 1970. Estimating the reliability, systematic error and random error of interval data. *Educational and Psychological Measurement*, 30(1):61–70.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270, San Diego, California. Association for Computational Linguistics.

Ji Young Lee, Franck Dernoncourt, and Peter Szolovits. 2018. Transfer learning for named-entity recognition with neural networks. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Bill Yuchen Lin and Wei Lu. 2018. Neural adaptation layers for cross-domain named entity recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2012–2022, Brussels, Belgium. Association for Computational Linguistics.

Xiaofan Lin. 2003. Impact of imperfect ocr on part-of-speech tagging. In *Seventh International Conference on Document Analysis and Recognition, 2003. Proceedings.*, pages 284–288 vol.1.

Zihan Liu, Genta Indra Winata, and Pascale Fung. 2020a. Zero-resource cross-domain named entity recognition. In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 1–6, Online. Association for Computational Linguistics.

Zihan Liu, Genta Indra Winata, Peng Xu, Zhaojiang Lin, and Pascale Fung. 2020b. Cross-lingual spoken language understanding with regularized representation alignment. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7241–7251, Online. Association for Computational Linguistics.

Zihan Liu, Yan Xu, Tiezheng Yu, Wenliang Dai, Ziwei Ji, Samuel Cahyawijaya, Andrea Madotto, and Pascale Fung. 2021. Crossner: Evaluating cross-domain named entity recognition. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 13452–13460. AAAI Press.

Margot Mieskes and Stefan Schmunk. 2019. OCR quality and NLP preprocessing. In *Proceedings of the 2019 Workshop on Widening NLP*, pages 102–105, Florence, Italy. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Juan Diego Rodriguez, Adam Caldwell, and Alexander Liu. 2018. Transfer learning for entity recognition of novel classes. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1974–1985, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Kepa Joseba Rodriquez, Mike Bryant, Tobias Blanke, and Magdalena Luszczynska. 2012. Comparison of named entity recognition tools for raw OCR text. In *Konvens*, pages 410–414.

Daniel van Strien, Kaspar Beelen, Mariona Coll Ardanuy, Kasra Hosseini, Barbara McGillivray, and Giovanni Colavizza. 2020. Assessing the impact of OCR quality on downstream NLP tasks. In *Proceedings of the 12th International Conference on Agents and Artificial Intelligence, ICAART 2020, Volume 1, Valletta, Malta, February 22-24, 2020*, pages 484–496. SCITEPRESS.

Amber Stubbs and Özlem Uzuner. 2015. Annotating longitudinal clinical narratives for de-identification: The 2014 i2b2/uthealth corpus. *J. Biomed. Informatics*, 58:S20–S29.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Myriam C Traub, Jacco Van Ossenbruggen, and Lynda Hardman. 2015. Impact analysis of ocr quality on research tasks in digital archives. In *International Conference on Theory and Practice of Digital Libraries*, pages 252–263. Springer.

Jing Wang, Mayank Kulkarni, and Daniel Preotiuc-Pietro. 2020. Multi-domain named entity recognition with genre-aware and agnostic inference. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8476–8488, Online. Association for Computational Linguistics.

Joey Tianyi Zhou, Hao Zhang, Di Jin, Hongyuan Zhu, Meng Fang, Rick Siow Mong Goh, and Kenneth Kwok. 2019. Dual adversarial neural transfer for low-resource named entity recognition. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3461–3471, Florence, Italy. Association for Computational Linguistics.

An Exploratory Study on Temporally Evolving Discussion around COVID-19 using Diachronic Word Embeddings

Avinash Tulasi
IIIT Delhi
NII Japan *
avinasht@iiitd.ac.in

Asanobu Kitamoto
NII Japan
kitamoto@nii.ac.jp

Arun Buduru
IIIT Delhi
arunb@iiitd.ac.in

Ponnurangam Kumaraguru
IIIT Hyderabad
pk.guru@iiit.ac.in

Abstract

COVID-19 has seen the world go into a lockdown and unconventional social situations throughout. During this time, the world saw a surge in information sharing around the pandemic, and the topics shared in the time were diverse. People's sentiments have changed during this period. Given the widespread usage of Online Social Networks (OSN) and support groups, the user sentiment is well reflected in online discussions. In this work, we aim to show the topics under discussion, evolution of discussions, change in user sentiment during the pandemic. We also demonstrate the possibility of exploratory analysis to find pressing topics, change in perception towards the topics, and ways to use the knowledge extracted from online discussions. For our work, we employ Diachronic Word embeddings which capture the change in word usage over time. With the help of the analysis from temporal word usages, we show the change in people's opinion on COVID from being a "conspiracy", to the post-COVID topics surrounding "vaccination".

1 Introduction

Analysing and estimating the impact of changes in social lives of people from time to time is key in digital humanities. With the advent of Online Social Networks (OSNs) it has become easy to access what people have been sharing their lives online, good or bad and seeking support in the online communities. During the pandemic, people seek help and they have also shared information with one another. To understand how people's lives were affected, and to contribute towards digital humanities, we look at conversations on COVID-19 during the pandemic. We listed the wide variety of topics discussed, and the support from each other when they were stuck with the illness.

COVID-19 pandemic has been a huge disaster for humanity. It has changed the way we live forever during the ensued lockdown from the pandemic. We saw a lot of community support and huge participation in supporting each other during the testing times. Similarly, COVID-19 is the first known global pandemic in the Internet age, when the information disposal is rapid and the ability to communicate anywhere on the earth takes seconds. It is of immense significance to see how people communicated around coordinating and how people supported each other during the times, as we can assess the ability of the world wide web in effect.

The information revolution in the form of worldwide web has also come with its own challenges like fake news, conspiracy theories, hate speech, etc. It is of particular interest to understand how the conspiracies and the narrative around COVID-19 has changed ever since the pandemic broke. For example, at the beginning of the pandemic, there has been a lot of speculation that COVID-19 is a hoax, and the virus is just political propaganda. Such narratives can be found around the world, including the United States, India, China, and Japan. However, it did not take long for people to realize that COVID-19 is, in fact, a pandemic. So, during the same period, people started discussing online about the pandemic and the narrative around conspiracy theories. In this work, we aim to understand how the topics involved in COVID-19 have changed since the inception of the pandemic till date.

For this work, we use a data set taken from Reddit, which is a popular social network built around smaller communities called subreddits. Particularly, we choose the subreddit r/COVID19positive [1]. Our data set contains all posts and comments made since the inception of the subreddit, and we use the data to understand the narratives around COVID-19 on how people's perception has

* The work was done as a part of an online internship program with NII Japan.

[1] https://www.reddit.com/r/COVID19positive/

Proceedings of the Workshop on Natural Language Processing for Digital Humanities (NLP4DH), pages 183–190
December 19, 2021.

changed. Specifically, we aim to assess the change in word usages on the basis of proximity among different words in the initial stages of the pandemic till date. The proximity of words and word usage is of importance because, togetherness of words gives us an idea on how people are thinking and what people are talking about. So, to estimate the word usage and proximity, we use robust Machine Learning methods such as Diachronic word embeddings (Kutuzov et al., 2018).

The Diachronic word embeddings (Montariol, 2021) take different embedding models trained on text corpora which are linked together in time. The text corpora are expected to evolve over time, and the temporal usage of words is captured by taking smaller snapshots of the corpus. To employ the technique, we train individual models and then align the models in such a way that the same words stay in relatively similar positions across time; thereby making the change in word embeddings relative. By closely looking at snapshots and the drift of words among snapshots, we can estimate the change in word usage. Multiple works have studied and seen a change in word usage throughout human history with the lens of corpora.

COVID-19 being two years old, we aim to do a word evolution estimation within these two years for our work. We split our corpus into month-long sub-corpora and train the embedding models on these individual corpora. Later, we align the corpora to maintain the word similarity across time. Once we align the word similarity across time, we estimate the divergence of words or drift from each other. The divergence and drift give us information on how the words have changed. Some interesting observations from our work include the usage of 'conspiracy'; at the beginning of the pandemic, the word and COVID-19 are closely related, and as time progresses, the word conspiracy disappears from the neighborhood of COVID. Similarly, COVID-19 is known for its symptoms such as cough and cold. During the initial phase, these words are seen away from 'COVID' or 'virus'. As time progresses and humanity goes into waves of huge infections, the words come together.

Humanity was lucky enough to produce vaccines in a short time, a narrative around vaccines and their evolution is also interesting to look at. Although a lot of vaccine-related discussions were taking place in the initial days of COVID, the usage of vaccines along with infection has increased recently (as of November - 2021). With our work, we also aim at developing a system that can be used by people with little to no technical knowledge of how word embeddings operate and obtain the information we extract.

In summary, our work focuses on :

1. Did narratives around COVID-19 and discussions around the pandemic have evolved over time? How?

2. The narrative around how getting 'vaccinated' and getting 'positive' are being used together? What does it imply for the community?

3. A system that can assist domain experts such as medical professionals, and journalists in accessing our findings in a visual form.

2 Related Work

In this section, we discuss the works that have studied Social Media websites during the COVID-19 crisis. In an early work, (Liu et al., 2020) the authors have collected Chinese articles about COVID-19 and reported the user sentiment around the pandemic. They have used News articles as the dataset. Similar works are seen in Brazil (de Melo and Figueiredo, 2021), that used Twitter and four European nations (Ghasiya and Okamura, 2021) based on news articles. We see a common pattern in the works which is to collect a publicly available dataset, use a language model and report the findings along side the real-world happenings.

Another widely observed theme in literature is to extract medical information from publicly available articles. The works (Murray et al., 2020; Sarker and Ge, 2021; Wu et al., 2021; Kumar et al., 2021) attempt at extracting medical information such as symptoms, post-COVID medical status, topics around vaccination using OSN data. The OSN of choice in the majority of these works is Reddit. We place our work at an intersection of the above-mentioned literature. We note the absence of literature on the evolution of themes, scientifically extracting and evaluating the change in discussions. Our work bridges the gap in the literature.

3 Methodology

For our work, to study the evolution in perception towards the pandemic related to COVID-19, we choose the subreddit r/COVID19positive as our

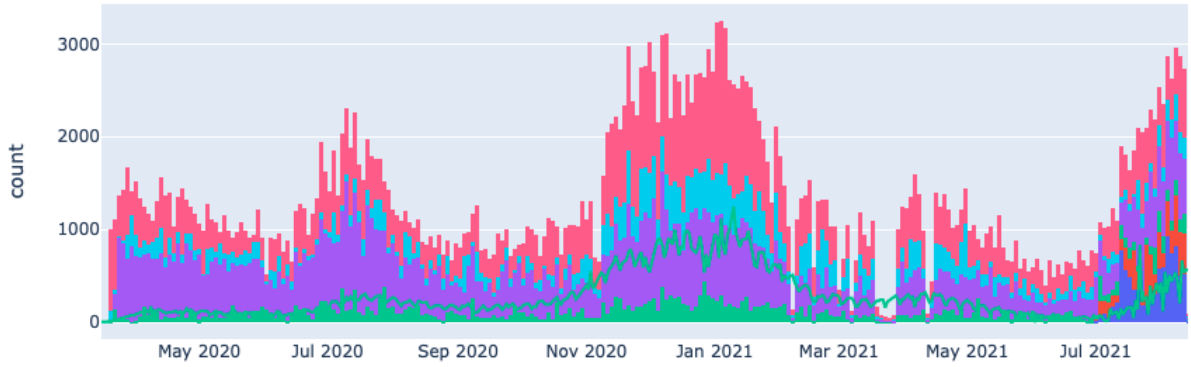

Figure 1: Number of posts per day in our dataset. The colors in the figure represent different flairs on r/COVID19positive subreddit. We can see the emergence of colors *red, green* which represent flairs related to *vaccines, tested positive* in early 2021. The figure also depicts the daily cases in the USA as a line. Our dataset closely follows the number of cases registered in the USA.

data source. Reddit is a community-centric social network where users form communities specific to topics and intents called subreddits. The r/COVID19positive subreddit is "A place for people who came back positive for COVID19 can share your stories, experiences, answer questions and vent!" – according to the subreddit description. As the subreddit contains discussions related to support, experience, and QnA by people tested positive, our data will be rich in capturing diverse contexts while being very specific to our topic of interest. With more than 124K users, the subreddit contains discussions ranging from users seeking support from the loss of a family person, to users discussing the geopolitical issues surrounding the pandemic. Next, we discuss the dataset preparation.

3.1 Dataset Preparation

Firstly, for our data collection, we used the Pushshift-API [2] to collect the ids of all post ever made on the subreddit. We have the earliest post dating back to [April - 2020], and the latest post on [October - 2021]. Once we have the post ids, we use the official Reddit API - PRAW [3] for collecting posts and comments. Our framework takes a post and recursively collects all the comments on a given post along with metadata such as upvotes, awards, flairs, etc.

Having collected all the posts, we first take a look at the topics being discussed in our dataset. On Reddit, moderators of a subreddit can define the topics, and these are similar to hashtags on networks like Facebook and Twitter. The topics are called *flairs*; just like hashtags flairs are displayed in colored boxes at the end of a post making the post *tagged under a topic*. So, we take the help of flairs, which are accepted by community and created by moderators; we identify the topics being discussed on the subreddit r/COVID19positive. By making flairs our choice for analysis at the initial stage, we are also avoiding assumptions on topics under discussion.

A list of all flairs and their meaning on the subreddit r/COVID19positive is as listed below:

1. **Tested positive - {Me, Long Hauler, Family}:** The four flairs related to testing positive. Each flair shows a special case where either the original poster is tested positive *me*, or a family member of the poster is tested positive *family*, or someone related (or themselves) to the poster is battling for a longer time than usual *long hauler*.

2. **Question - {medical, to those who tested positive}** The two flairs represent queries to the communities by users. The first flair *medical* represents questions related to scientific research, particularly vaccination related to COVID 19. While the flair *to those who tested positive* contains posts about someone's plight while they were battling with the infection. The questions result in discussions related to symptoms and conspiracy theories, as we will see further in the paper.

3. **Vaccine - {tested positive, discussion}** The two flairs are related to vaccines and vaccination. These flairs appeared in the later part of

[2] https://github.com/pushshift/api
[3] https://praw.readthedocs.io/en/stable/

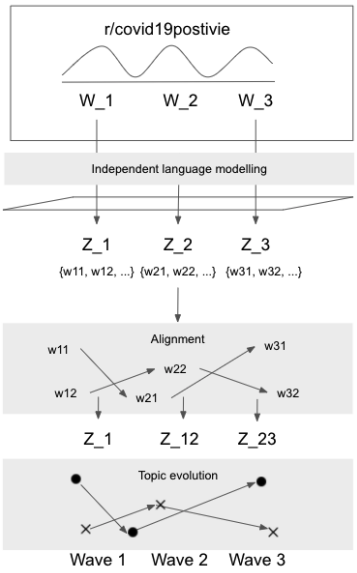

Figure 2: Framework used for our work. We have four distinct steps, where we split the dataset, then we obtain independent language models on the split corpora. Later we align the word embeddings to estimate and present the drift of each word as the final outcome.

the pandemic as shown in Figure 1. The flairs represent a changed infection pattern where people are seeing infections among the vaccinated population. Also, the vaccine-related discussion contains posts around and about different vaccines, vaccination strategies, and conspiracy theories related to vaccines.

The pandemic has a different infection rate in different nations owing to the local laws, control among the population, and other governing policies. We find that majority of our dataset contains posts made by people living in the United States of America, based on the occurrence of geography names. So, we plotted the infection rates in the USA on top of our daily posts, and the figure 1 shows the correlation between infections ranging in the USA and the posts being made on the subreddit r/COVID19positive.

Moving further in our study, we will now focus on the change in perception, the context of discussions, and key focus points with time during the pandemic. So, as our dataset is closely related to the USA infection statistics, we split the entire timeline into three parts, with the part 1 being from [Mar - Oct 2020] - where we see a dip in the infection numbers; part 2 from [Nov 2020 - Apr 2021], as we see another short dip in the infection numbers, and the remaining discussions are our part 3 which contains data from [May 2021 - till date]. Another rationale behind making the split is to capture the new emergence of infections among the vaccinated population, we make the third split when the use of flairs *vaccinated - tested positive* increase along with the infections numbers in the USA.

Once we make the split, we now have three corpora of our dataset. Additionally, we consider the posts and comments as independent documents because we are studying the overall evolution of topics during the pandemic. We call the three corpora C_1, C_2, C_3. The corpora contain special symbols, emoticons, etc. As a standard NLP pre-processing practise, we do lemmatization, cleaning of text with regular expressions, etc. Our resulting dataset contains 95,799 documents in C_1; 46,529 documents in C_2 and 55,696 documents in C_3. After extracting the common words being used in all of the three corpora, we work with 9,805 words. Having prepared the data and given an overview of the entire dataset and the potential topics in our dataset; we now proceed to discuss the framework using which we will answer our research questions.

3.2 Algorithm

With the dataset prepared, we now proceed to introduce Diachronic word embeddings (Montariol, 2021) and further processing of the social media data.

3.2.1 Introduction to Diachronic Word Embeddings

Traditional embedding models, when trained independently, do not capture information or knowledge present in another word embedding model. If a Corpus is evolving over time, then taking a word embedding or training a model on snapshots is not enough to capture the temporal evolution. By training multiple word models, we can capture the smaller snapshots and differences in nuances of the words in short intervals. However, these models are not comparable with subsequent models in a different time slice. To overcome the difficulty and ensure that words capture the meaning over time among multiple models, researchers have proposed Diachronic word Embeddings.

One way to overcome the challenge (which we adapt in our work) (Szymanski, 2017; Hengchen and Tahmasebi, 2021) is to align words among different models which were independently trained; keeping similar words in similar positions. The constraint being, words in subsequent models should

be aligned to maintain the neighborhood. By aligning models, we can ensure that words do not drift far away and that the difference between words is comparable across independent word models. For the alignment, we use pearson's distance as proposed in the original work.

3.2.2 Interpretation

The resulting word embeddings capture the temporal evolution of word usage. However, the evolution can be represented in the form of distance movement and neighbourhood change. As time progresses, the words keep moving in the latent space; with the movement the distance between words among subsequent timestamps may be large, or it may be small. If the distance is large, it means the words have changed their meaning significantly; if it is small, the words have stayed mostly the same. Additionally, another way to capture the temporal evolution of words is to use the neighborhood of words. Neighborhood reflects the context in which a word is used. Hence, by looking at the neighborhood, we understand what other words are more likely to co-occur with the given word. Using a similar approach, we study the contextual difference resulting from the contextual evolution of a given word over time. In this work, we employ these two methods and we present the results using the distance as well as the neighbourhood.

3.2.3 Representation Extraction

As discussed in an earlier Section 3.2.1 Diachronic word embeddings capture the changes in word usage over time. To answer our questions, we need to estimate how some words might drift away from other words as time progresses, indicating an evolution or change in user sentiment, topics of choice with time. With our framework, we apply the word embedding techniques, and then we align words w.r.t existing Diachronic embedding methods. We then discuss the implications of word drift and enhance the results with the use of *flairs* from earlier. Then we present our findings in detail. Now, we proceed to present the framework in detail.

Firstly, we train three independent word embedding models. For this part, we choose word2vec. Each corpus C_i gives an embedding model represented by Z_i, and the word representation in each embedding model is given by $w_{i,j}$ where i is the model and j is the word. Secondly, we proceed to align the word embeddings. For this part, we choose the method used by (Huang and Paul, 2019). The alignment is done pair-wise. We take the C_1, C_2 pairs, and we find the common vocabulary C_{12} among the two corpora. Then, we sort the words in the common corpus by the frequency of occurrences and align the words in the decreasing order of frequency in the target corpus. We then perform an SVD operation between the embedding matrices Z_1, Z_2 which only contain the common vocabulary. As a result of the operation we obtain an aligned embedding matrix Z_{12}. Similarly, we align other pairs of embedding matrices to obtain Z_{23}. Once the aligning operations are complete, we have three sets of embeddings Z_1, Z_{12}, Z_{23} that represent each wave, respectively. The entire framework is shown in Figure 2.

3.3 Representation of temporal word evolution

Now that we have created the three aligned word embedding models, with the help of Diachronic Embedding framework (NTAM), we have all the words in comparable positions. Particularly, with the help of plots and target keywords, we will decipher the user sentiment across the subreddit and the topic evolution from time to time. For making the plots, we take the position of different words in the three models, and then we also obtain their 10 nearest neighbours. As discussed in an earlier Section 3.2.1, neighbourhood and the near/far movement of words are key indicators of capturing temporal changes in topics.

We present the temporal evolution of words in our dataset in the form of two visual representations:

- **Neighborhood evolution** To assess the change in the context of word usage, we plot the words from each model on a single 2D plane. As mentioned in section 3.2.1 the neighborhood represents the context of word usage, and words in the neighborhood. These neighborhood words are more probable to occur co-occur with the base word, making them relevant. Similarly, the closeness of a tight-knit neighborhood shows a distinctive topic, and a neighborhood in which the distances are not tight-knit can be attributed to a neighborhood disintegrating. In our work, we plot the neighbourhood of a given word. For interpretation, if the neighborhood is small, we want the readers to understand that the topic

is concrete and the word usage belongs to a single context. Similarly, if the neighborhood is large, the reader should understand that the word is drifting away from a given neighborhood, and the frequency of word usage has decreased.

- **Word Drift** Another key aspect in studying the difference in the temporal evolution of topics is the word drift. "How much does a word move for away from itself over time?". By comparing two different words and the drift along time, we can show how the usage of the word has changed comparative to another word. In our work, we look at 'cancer' and 'cold' for example. Cancer was not widely used at the beginning of the pandemic however it started being frequent post pandemic. By showing the difference between words, we highlight the word importance in the global context.

Figures 3, 4, 5 show the difference between different words as seen in the neighborhood and the drift.

4 Results and Discussion

Having established the methodology and visualizations that aid us in understanding word evolution, we now discuss the results.

4.1 Narrative evolution during COVID-19

Conspiracy: Sentiments and discussions around COVID-19 have seen a vast change. Firstly there was some speculation that the virus outbreak could be a hoax, and governments are trying to push propaganda with a narrative around the Coronavirus. This has resulted in a lot of conspiracy theories and agenda against the government. However, during the later parts, particularly after the first wave of infections, people have started believing that COVID-19 is indeed a real threat. The later parts show a different sentiment among the users. With the help of words drifting apart from each other and by looking at the membership of words such as "conspiracy", "government" and "hoax", we demonstrate that there is indeed a change in the neighborhood among these words. The change in the neighbourhood also indicates that people have moved away from believing in a hoax. An increase in the popularity of scientific articles that have been shared and discussed on the COVID-19

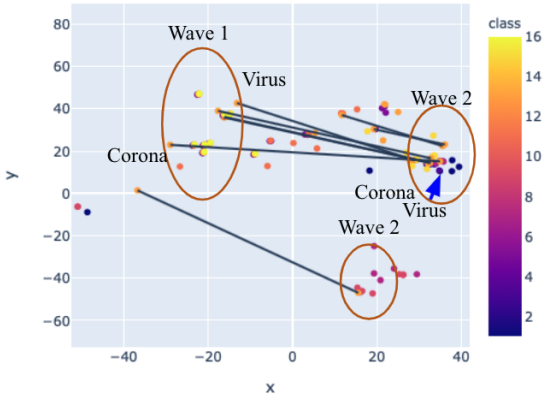

Figure 3: Figure showing the difference between wave 1, and 2. The words in the picture are 'corona', 'pandemic' 'virus', etc.

positive subreddit is also present. We show the nearest neighbours of conspiracy words before the first year and later in the second wave in Figure 3.

Symptoms: As we have seen, discussions around conspiracy have dropped down; also, conversations saw a surge in discussions around symptoms. In this part, we study the COVID-19 symptoms such as *cough, fever*. We show that these words were very close to each other in the second half of our time split(C_2). However, as time progressed, the symptoms are seen very close to the words such as *COVID, coronavirus* and *infection* which shows that users are thoroughly using symptoms along with discussions key to the pandemic. The key change to notice during the third wave of infections (C_3) is the symptoms being nearest-neighbors to chronic diseases such as *cancer* and *heart failure*. This change is particularly important because, even though patients are tested negative; for patients who have recovered from a COVID-19 infection, there is a chance of potential hazard. We show the the neighborhood change around words representing symptoms in Figure 4.

Post-COVID Symptoms: Chronic diseases like *cancer* and *heart failure* have seen a surge in usage as well as they have moved closer to the keywords. The change can be attributed to the fact that people with pre-existing conditions are more vulnerable to COVID-19. A long-standing battle with the disease is correlated with a lot of deaths reported after tested negative; with a chronic existing condition. We have also seen the closeness of word alongside these diseases, which is once again factor that plays an important role in the community health. Old people being a potential target

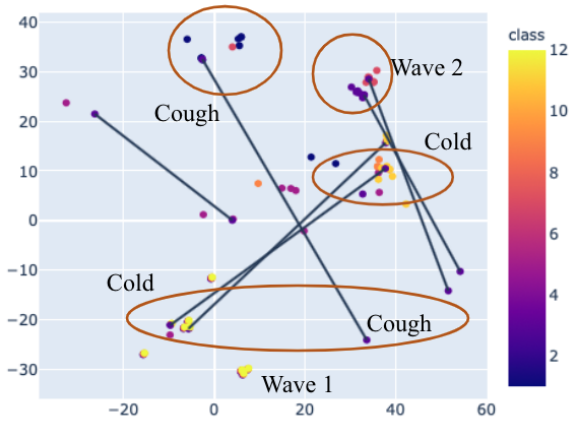

Figure 4: Figure showing the difference in symptoms during the pandemic. The evolution of symptoms here shows a change in user perspective towards the same.

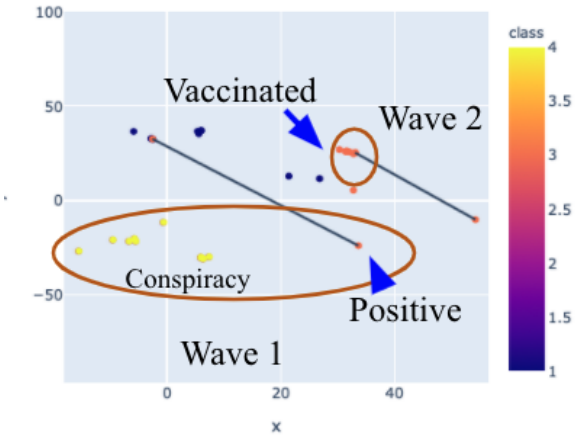

Figure 5: Figure showing the discussions around vaccinated and getting positive.

for COVID-19, could not recover completely from the infection and eventually getting some other disease was seen. we show the neighborhood change of COVID symptoms post-COVID symptoms in Figure 5.

4.2 Vaccinated - tested positive

Having seen the comparison between the first and the second wave and studied how word usage has changed; we have also established - during the same time, there is a surge in usage of words such as cancer and heart failure. However, another important aspect to consider is the possibility of getting tested after being vaccinated. We have captured the phenomenon in our data set. With the help of flairs we now study how the post-pandemic and the post-COVID community is dealing with getting vaccinated.

Getting vaccinated does not make a person immune to the infection. But, the infection will not have a long-standing effect on a person. Vaccination is there to eradicate the effect of COVID. In the United States, a lot of people have been vaccinated, and they are also getting an infection. So, in this context, we look at the *vaccinated* flair, and we find that a lot of people are recovering within their homes; for example, refer to the link [4]. Similarly, post-pandemic syndrome effects are very clear in old people and people with long-standing health conditions. The same is reflected in post with flair "tested positive family". Please note that the flair is representative of a family being infected. In this flair, particularly during the post-pandemic period we find that a lot of people are getting tested positive and being admitted for long-standing diseases such as *cancer* or *heart disease*. Old people in particular are getting infected after getting vaccinated.

With this subsection, we shed light on the future of COVID-19 pandemic after the wave 3. We can expect the vaccinations to stand as a guard against a flood of infections. However, we cannot expect infections to go down. Similarly, people with bad health conditions are more vulnerable even after getting vaccinated.

5 Concluding Remarks

In this work, we have used Diachronic word embeddings to assess the temporal evolution of topics around COVID-19 pandemic. For one and half years, we take the help of r/COVID19positive subreddit. In order to conduct our study, we have captured the users throughout the three waves of the United States. We have closely followed the pandemic in the United States because our conversations have followed the peaks from the United States. In our work, we split the Corpus into three, and compared the three sub-corpora; each subcorpora capturing a different phases of the pandemic. In the first wave, we've had a lot of uncertainty around COVID, and a lot of conspiracy theories were floating around. We find that by the second wave, people got serious, and there was a lot of support going on in the community. People were discussing vaccines in the latter half of the second wave. However, during the third wave, we have seen that people were discussing being vacci-

[4] https://www.reddit.com/r/COVID19positive/comments/qu26z4/tested_positive_twice_in_4_months_fully_vaxxed/

nated and getting a positive result on their COVID test after vaccination. We also demonstrate an important aspect of post-COVID symptoms, which are related to long-standing illness such as *cancer* and *heart failure*. With this work, we demonstrate the utility of Diachronic embeddings in the real world. We also demonstrated the application of Diachronic embeddings on a very short time period unlike any previous literature or research.

In the future we aim to build a dashboard that can be easily used by domain experts, medical professionals and journalists to access the information we extracted in our work.

Acknowledgement

The authors would like to thank National Institute of Informatics (NII), Japan for providing the online internship opportunity. Also, the authors found the thesis (Montariol, 2021) extremely resourceful for their research.

References

Piyush Ghasiya and Koji Okamura. 2021. Investigating covid-19 news across four nations: A topic modeling and sentiment analysis approach. *IEEE Access*, 9:36645–36656.

Simon Hengchen and Nina Tahmasebi. 2021. A collection of swedish diachronic word embedding models trained on historical newspaper data. *Journal of Open Humanities Data*, 7.

Xiaolei Huang and Michael J. Paul. 2019. Neural temporality adaptation for document classification: Diachronic word embeddings and domain adaptation models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4113–4123, Florence, Italy. Association for Computational Linguistics.

Navin Kumar, Isabel Corpus, Meher Hans, Nikhil Harle, Nan Yang, Curtis McDonald, Shinpei Nakamura Sakai, Kamila A Janmohamed, Weiming Tang, Jason L Schwartz, et al. 2021. Covid-19 vaccine perceptions: An observational study on reddit. *medRxiv*.

Andrey Kutuzov, Lilja Øvrelid, Terrence Szymanski, and Erik Velldal. 2018. Diachronic word embeddings and semantic shifts: a survey. *arXiv preprint arXiv:1806.03537*.

Qian Liu, Zequan Zheng, Jiabin Zheng, Qiuyi Chen, Guan Liu, Sihan Chen, Bojia Chu, Hongyu Zhu, Babatunde Akinwunmi, Jian Huang, et al. 2020. Health communication through news media during the early stage of the covid-19 outbreak in china: digital topic modeling approach. *Journal of medical Internet research*, 22(4):e19118.

Tiago de Melo and Carlos MS Figueiredo. 2021. Comparing news articles and tweets about covid-19 in brazil: sentiment analysis and topic modeling approach. *JMIR Public Health and Surveillance*, 7(2):e24585.

Syrielle Montariol. 2021. *Models of diachronic semantic change using word embeddings*. Ph.D. thesis, Université Paris-Saclay.

Curtis Murray, Lewis Mitchell, Jonathan Tuke, and Mark Mackay. 2020. Symptom extraction from the narratives of personal experiences with covid-19 on reddit. *arXiv preprint arXiv:2005.10454*.

Abeed Sarker and Yao Ge. 2021. Long covid symptoms from reddit: Characterizing post-covid syndrome from patient reports. *medRxiv*.

Terrence Szymanski. 2017. Temporal word analogies: Identifying lexical replacement with diachronic word embeddings. In *Proceedings of the 55th annual meeting of the association for computational linguistics (volume 2: short papers)*, pages 448–453.

Wei Wu, Hanjia Lyu, and Jiebo Luo. 2021. Characterizing discourse about covid-19 vaccines: A reddit version of the pandemic story. *arXiv preprint arXiv:2101.06321*.