Language Models Lack Temporal Generalization and Bigger is Not Better

Stella Verkijk
Huygens Institute
Vrije Universiteit Amsterdam
s.verkijk@vu.nl

Piek Vossen
Vrije Universiteit Amsterdam

Pia Sommerauer
Vrije Universiteit Amsterdam

Abstract

This paper presents elaborate testing of various LLMs on their generalization capacities. We finetune six encoder models that have been pretrained with very different data (varying in size, language, and period) on a challenging event detection task in Early Modern Dutch archival texts. Each model is finetuned with 5 seeds on 15 different data splits, resulting in 450 finetuned models. We also pre-train a domain-specific Language Model on the target domain and fine-tune and evaluate it in the same way to provide an upper bound. Our experimental setup allows us to look at underresearched aspects of generalizability, namely i) shifts at multiple places in a modeling pipeline, ii) temporal and crosslingual shifts and iii) generalization over different initializations. The results show that none of the models reaches domain-specific model performance, demonstrating their incapacity to generalize. mBERT reaches highest F1 performance, and is relatively stable over different seeds and datasplits, contrary to XLM-R. We find that contemporary Dutch models do not generalize well to Early Modern Dutch as they underperform compared to crosslingual as well as historical models. We conclude that encoder LLMs lack temporal generalization capacities and that bigger models are not better, since even a model pre-trained with five hundred GPUs on 2.5 terabytes of training data (Conneau, 2019) underperforms considerably compared to our domain-specific model, pre-trained on one GPU and 6 GB of data. All our code, data, and the domain-specific model are openly available.[1]

1 Introduction

Generalizability is a vital aspect of machine learning. A model learns patterns from its training data that it should be able to apply to data it has not seen before and might differ in some aspects. Since the rise of deep learning, black box models and 'big data' in Natural Language Processing, knowing what generalizibility exactly is and how to study it has become problematic (Hupkes et al., 2023).

One aspect of generalizability is generalization over domains and genres; a model trained to detect events in newspaper articles should also be able to do so in tweets. A more specific type of domain shift is temporal shift: where the training and test data come from different periods. Temporal shifts are complex as they involve different styles and even different entities and events. These days, vast amounts of contemporary data are available, but systems trained on these data do not always work well on historical data (Manjavacas and Fonteyn, 2022a), not only because of orthographical and syntactic differences, but also because of semantic shift (Kutuzov et al., 2018; Hamilton et al., 2016).

Event detection is a well-studied task and ex-

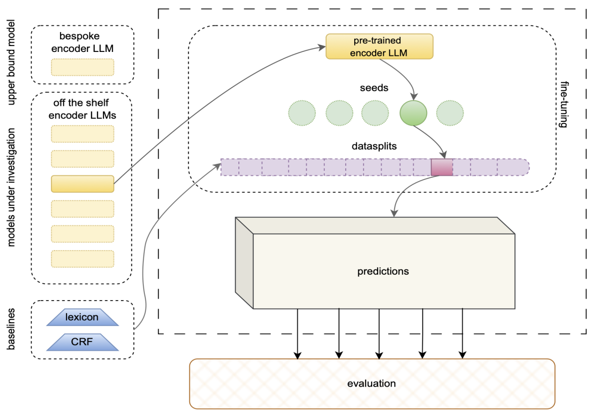

Figure 1: Experimental set-up: each encoder goes through a process where it is initialized 5 times with different seeds and fine-tuned and tested on a different datasplit. This enables us to i) evaluate different aspects of generalizability and ii) control for randomness in the finetuning process when comparing base models' downstream performance by taking the average performance over all seeds and datasplits.

[1]See our repo for code and data and our huggingface page for the model.

Findings of the Association for Computational Linguistics: ACL 2025, pages 20629–20637
July 27 - August 1, 2025 ©2025 Association for Computational Linguistics

tremely useful for many different areas of society, such as news digestion (Vossen, 2018), clinical decision making (Zhang et al., 2020) and historical research (Sprugnoli and Tonelli, 2019). It is a complex task and performance on this task has been shown to be prone to suffer from situations where the train and test data come from different domains (Hong et al., 2018).

Hupkes et al. (2023) draw out a taxonomy of various aspects of generalizability in their GenBench initiative and highlight understudied regions. They point out that the vast majority of studies focus on shifts between training (finetuning) and testing, not considering shifts between pre-training and training. Furthermore, a comparatively small percentage of studies investigate shifts in multiple stages of the modeling pipeline. Hupkes et al. (2022) also point out that only a limited number of tasks investigate temporal generalizability, none including event detection.

The GLOBALISE project[2] is building software that allows historians to search through the archives of the Dutch East India Company (Verenigde Oostindische Compagnie; VOC): a corpus of over 5 million handwritten pages from the 17th and 18th centuries containing valuable information on the history of early colonialism. The corpus has been transcribed and partially annotated for event extraction. The data provide an excellent use-case to study temporal generalizibilty.

In this study, we finetune several encoder Large Language Models (LLMs) on this newly defined event detection task in Early Modern Dutch archival texts. We finetune models trained on contemporary Dutch (BERTje (De Vries et al., 2019) and RobBERT (Delobelle et al., 2020)), a model trained on Dutch from 1500 to 1950 (GysBERT (Manjavacas and Fonteyn, 2022b)), a different version of that model for which the VOC corpus was added to its pre-training data (GysBERT-v2), and multilingual models (Devlin et al., 2019; Conneau, 2019). In doing so, our study provides a unique opportunity to investigate shifts in pretrain-train scenarios as well as finetune-test scenarios, investigating temporal generalizability as well as cross-lingual generalizability. To the best of our knowledge, we are the first to study temporal generalization with respect to event detection.

Additionally, we address the issue of generalization of models over different initializations, something GenBench does not explicitly take into account. Stochasticity is a known problem in NLP (Bender et al., 2021; Khurana et al., 2021). If a pre-trained model finetuned ten times produces one good model but nine mediocre ones, what does that say about the level of generalizability the pre-trained model carries? Our study considers this by comparing consistency between models finetuned with different seeds.

[2] https://globalise.huygens.knaw.nl

2 Related work

Earlier literature in NLP shows great effort at creating datasets for event detection in English (Walker et al., 2006; Saurí et al., 2006; Pustejovsky et al., 2010; UzZaman et al., 2013; Cybulska and Vossen, 2014; Styler IV et al., 2014; Bethard et al., 2016). Since then many systems were built to tackle this task, from feature engineering techniques (Ji and Grishman, 2008) to deep learning (Li et al., 2022).

To improve scores by just a few decimals (68.0 F1 for a finetuned RoBERTa vs. 68.5 for a more complex approach (Wang et al., 2021)), recent event detection systems involve heavy engineering, for example including graph structures (Nguyen and Grishman, 2018) or ensembling various types of modules into a pipeline (Zhang et al., 2024). Most of these methods have been tweaked to work well on the most popular benchmarks - ACE (Walker et al., 2006) and MAVEN (Wang et al., 2020) - but they remain unable to generalize to other datasets (Wang et al., 2021).

Machine learning systems have been shown to heavily underperform when tested out of domain in many fields (Gulrajani and Lopez-Paz, 2020). In recent years, the field of NLP has also started to raise concerns about perfecting systems on benchmarks, showing how models that reach excellent performance on certain train/test splits fail on simple challenge examples and commit errors in real-world scenarios (Kiela et al., 2021; Plank, 2016), indicating they may rely on stereotypes and memorization (Hupkes et al., 2023). This suggests that generalization by NLP models is often overestimated (Ribeiro et al., 2020). Now, scholars experiment with prompting generative LLMs in a zero-shot fashion for event detection, but the results vary wildly and are unreliable (Kristensen-McLachlan et al., 2023; Gao et al., 2023). To better understand what generalization means and to make progress, we should investigate what models can or cannot learn when it comes to event detection.

Domain-specific pre-training of encoder LLMs has shown to improve downstream performance in various domains (Lamproudis et al., 2022; Chalkidis et al., 2020; Müller et al., 2023; Verkijk and Vossen, 2021). It has also been shown that the crosslingual capacities of multilingual LLMs can work well for Early Modern Dutch (Arnoult et al., 2021). However, no work has been done yet that thoroughly compares the performance of contemporary, historical and multilingual models on historical text to test their generalizability.

	Data	Param	tok/byt
GysBERT	H	110M	7.1B /
GysBERT-v2	HV	110M	8.3B /
BERTje	C	109M	2.4B / 12GB
RobBERT	C	117M	6.6B / 39GB
mBERT (base)	M	179M	/
XLM-R (base)	M	279M	/ 2.5TB
GloBERTise	V	117M	/ 6GB

Table 1: Models with types of language present in and volume of pre-training data (H = historical; V = VOC; C = contemporary; M = Multilingual). All info missing in this table could not be found in the relevant papers.

3 Methodology

3.1 Data

GLOBALISE is a multidisciplinary effort to develop a (re)search interface for the archives of the VOC. This is a corpus of over 5 million (scans of) handwritten pages from the 17th and 18th centuries describing practices of trade, colonization and politics. These scans go through a specialized Handwritten Text Recognition pipeline[3] before any further processing. The imperfect HTR performed on a version of Dutch from before there were strict writing conventions results in much more noisy data than LLMs are usually pre-trained on. Also, the differences in language between Early Modern Dutch and contemporary Dutch are considerable (Verkijk et al., 2024).

3.2 Task

We finetune LLMs on an event detection task specifically defined for GLOBALISE. Through interdisciplinary collaboration , guidelines were developed for the extraction of events deemed relevant for conducting historical research on this source (Verkijk and Vossen, 2023). This resulted in an annotation scheme encompassing around 80 event types.[4] Note that the goal is thus to teach systems not to label every predicate they encounter, but only those that are relevant according to the scheme, which are sometimes quite common, like *Transport*, and sometimes more typical of the domain and time, like *Mutiny*. Additionally, events get annotated both when there is a direct reference to an event class ('the ship left' referring to *Leaving*) as well as an indirect reference ('the king's widow' referring to *BeingDead*). For the sake of our current goal of comprehensively evaluating temporal generalizability, we only evaluate on event detection and not on classification, i.e. a binary token classification task indicating whether a token does or does not refer to an event. We expect the event concepts as such to be relatively stable over time, but to be associated with different world entities. Models pretrained on contemporary data thus should still be able to generalise and apply these concepts to the 'old world'.

The data we finetune on, introduced in Verkijk et al. (2024), was annotated by 3 teams of 2 annotators; all specialized historians trained at the task. The data contain (parts of) 15 different documents, comprising a total of 107 handwritten pages/scans. The longest annotated text is 18 pages and the shortest 1. For four of the documents the inter-annotator agreement for event detection is 71% and for the rest 84% (IAA calculated per annotation round).

3.3 Models

We finetune and test six models that include some form of Dutch in their pre-training data. They differ in architecture (RoBERTA vs. BERT), size, and in the data they were pre-trained on, being more and less similar to the data we finetune and test on. We differentiate between contemporary Dutch, historical Dutch (Dutch from anytime before the 20th century) and VOC Dutch: transcribed Early Modern Dutch (1600-1800) as written in the archives of the VOC. The latter is the domain that we finetune and test on. See Table 1 for an overview.

We pre-train a new Transformer encoder on around 5 million scans of pages from the Archives of the VOC (6GB of text data).[5] The model, which we name GloBERTise, is trained from scratch on only domain-specific data. It is a RoBERTa-based model, trained for two epochs on one GPU[6]. This model functions as our upper bound model: if any of the other models reaches similar performance,

[3]Loghi: `https://github.com/knaw-huc/loghi`

[4]See the annotation guidelines and the event wiki

[5]The complete dataset is available on this Dataverse page

[6]See our repo for code and documentation for pre-training

its ability to generalize and thus perform well in the target domain is demonstrated.

3.4 Experimental set-up

Our experimental setup is built along three axes: i) variation in the pre-trained models we use, ii) variation in the data we finetune and test on and iii) variations in the seeds we use when finetuning. See Figure 1 for an overview. We split the data on document level and part it in 15 folds: for each fold, one document in the dataset was set apart for testing (following a so-called leave-one-domain-out cross-validation scheme (Gulrajani and Lopez-Paz, 2020)). This way, we can see whether models perform worse on documents from earlier times. We use five seeds. Models are finetuned on a binary token classification task, indicating for each token whether it refers to an event or not, i.e. the *None* class is overrepresented. On average, 8.3% of tokens refers to an event.

All models were finetuned for 5 epochs with a learning rate of 5e-5. See Table 8 (Appendix) for further parameter settings. We compare models' performances to two baselines: a lexical approach and a Conditional Random Forest (CRF) algorithm. The lexicon was created through an iterative process of annotation analysis, expert input and a domain-specific word2vec model (trained solely on the VOC corpus). The CRF was trained with word embeddings of the same word2vec model.[7]

4 Results & Discussion

4.1 Generalization over shifts between pre-training and finetuning

Table 2 shows scores per model averaged over data folds and seeds. Highest scores are indicated in boldface, runner-up scores are underlined. Mention accuracy was calculated as follows: if one or more of the tokens within a gold event mention span (i.e. "ordonnantie" in "d'ordonnantie ende last") is recognized as an event token in the predictions, we see it as overlap.

None of the models reaches the performance of our domain-specific model. mBERT performs best overall, with GysBERT closely following. Models trained only on contemporary Dutch score lowest. The difference between GysBERT and GysBERT-v2 is small. The latter scores highest in precision after the lexical approach. The difference in recall scores between models is smaller.

The difference in performance between the two multilingual models tested is noteworthy for the following reason: mBERT performs better even though XLM-R is the bigger model, with more parameters and trained on more training data: XLM-R used all of mBERT's training data (WikiPedia) and added CommonCrawl to it.

Our results show that multilingual models have potential in scenarios involving temporal shift. This however might depend on the language, domain and time of origin of the data involved. Early Modern Dutch has similarities to English, both being West-Germanic, making it perhaps prone to benefit from crosslingual transfer with English. To get more insight into this, we finetune and test three English models as a control-case: BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019b) and MacBERTh (Manjavacas and Fonteyn, 2021), a model pre-trained on data from 1450-1950. Tables 5 to 7 show that all English models perform significantly worse. For two out of five seeds, BERT does not predict any event. The fact that mBERT performs much better than English models strengthens the conclusion that the crosslingual capacities of a multilingual model help it to generalize to data from a different period and domain.

Our results indicate that the VOC domain is so specific that the GysBERT models are not representative. A domain is a mixture of domain, genre and time, and the types of documents in the GysBERT models' pre-training data mostly represent a very different genre, namely that of newspapers, books, journals and literature, and span almost 5 centuries.

4.2 Generalization over initializations

Table 3 shows the standard deviation of the separate models between the seeds they were initiated with. We first calculated the standard deviation for each specific data split and then averaged those, as opposed to taking the average performance of a model on all data splits per seed and calculating standard deviation between those averages. XLM-R, the model with most parameters and training data, shows highest standard deviation and thus stochasticity by far, followed by RobBERT (117M param), both scoring higher standard deviation scores than mBERT (179M param).

Our domain-specific model, also a RoBERTa architecture, performs best but is not less stochastic than mBERT. This seems to indicate that it is not necessarily size (amount of parameters) that makes models more stochastic, but architecture.

[7] For more info on the lexicon see our repo and our blogpost

	GysBERT	GysBERT-v2	XLM-R	BERTje	RobBERT	mBERT	GloBERTise	CRF	lexicon
P-event	0.69	0.71	0.62	0.64	0.63	0.63	0.69	0.57	**0.83**
R-event	0.40	0.39	0.39	0.36	0.37	0.43	**0.50**	0.30	0.22
F1-event	0.49	0.48	0.46	0.45	0.45	0.50	**0.56**	0.39	0.34
mention acc.	0.55	0.53	0.52	0.47	0.50	0.57	**0.64**	0.40	0.34

Table 2: Scores on detecting events averaged over data folds and seeds

	GysBERT	GysBERT-v2	XLM-R	BERTje	RobBERT	mBERT	GloBERTise
P-event	0.037	0.035	0.046	0.036	0.036	0.034	0.030
R-event	0.029	0.026	0.043	0.026	0.033	0.027	0.029
f1-event	0.027	0.022	0.037	0.021	0.030	0.021	0.021

Table 3: Standard deviation scores between seeds for each model

4.3 Generalization over shifts between finetuning and testing

Looking at the variation in performance between datasplits and hence period in time, there is no clear pattern. Table 4 (Appendix) shows performance per datasplit of our domain-specific, the worst and the best performing model. The variation in performance is not negligable: BERTje's scores vary between an F1 of .29 and .62. mBERT has slightly less variation between datasplits, scoring between .35 and .64. GloBERTise also shows high variance, scoring between .40 and .72. All three models score worst on the document from 1713 and best on that from 1707. Interestingly, both these documents were annotated in the same annotation round and both feature a high event density. The cause of these performance differences per split thus remains unclear (see Section 5).

5 Avenues for Future Work

Many of the findings in this paper need further investigation to be explained. As mentioned, we find that the GysBERT models are not representative of the VOC domain. However, the performance may also be lower because they were based on the uncased version of BERT or because they performed quality filtering of the pretraining data, discarding very noisy data. We consciously did not perform any data filtering for GloBERTise in order to represent the noise of the domain. Since there is very little documentation available for the GysBERT models, it is hard to study these hypotheses further.

It remains unclear why mBERT outperforms XLM-R. It might be the case that the next sentence prediction (NSP) objective during pre-training, which teaches the model a form of topic modelling (Lan et al., 2019), proves helpful in a topic-dependent event detection task like ours. Similarly, it might be worthwhile to investigate what makes BERT models less stochastic than RoBERTa models. Again, it might be the inclusion of NSP, but it could also be due to the different input formatting (that refrains from using two sentences) or RoBERTa's larger batch size.

Further research could also look into the differences in performance depending on the datasplit in finetuning/testing. Since none of the metadata show correlation with the scores per split, the subject matter of the tested document might be the deciding factor in the performance of the model. This deserves more attention.

6 Conclusion

We have shown that encoder LLMs lack the ability to generalize to different domains and different periods in history. None of these models, including models that have Dutch in their pre-training data, comes close to a domain-specific model pre-trained on only 6GB of data on an event detection task in Early Modern Dutch.

We do not find a clear correlation between stability over seeds and datasplits and overall performance, but we find that RoBERTa models tend to be more stochastic than BERT models. Multilingual models are more capable of temporal generalization than single-language models. Of the two multilingual models evaluated, the smaller mBERT outperformed the larger XLM-R. Contemporary single-language models are shown to be least capable of generalization compared to single-language models that included data from before the 19th century. However, even historical models that were not adapted to the specific domain of the archives of the VOC company underperformed.

Our research re-iterates that building language technology that takes the specifics of a domain into account will outperform general models, even if they are much larger.

7 Limitations

In our set-up, we do not experiment with different learning rates and epochs in order to keep results comparable. Hence, using different parameters during finetuning might have an impact on the results. The findings might also differ for a different use case, i.e. focusing on a different task or a different language variety or domain. For example, mBERT's performance might worsen compared to XLM-R when finetuning and testing on a language in a different script, because of its Unicode character based tokenizer compared to RoBERTa's byte-level BPE tokenizer (Tufa et al., 2024).

Acknowledgements

This research was funded by the Dutch Research Council (NWO) under project number 175.2019.003 and is part of the GLOBALISE project. All finetuning and pre-training was done on the National Supercomputer Snellius, which we were granted access to by the Dutch national e-infrastructure with the support of the SURF Cooperative using grant no. EINF-11854.

References

Sophie I. Arnoult, Lodewijk Petram, and Piek Vossen. 2021. Batavia asked for advice. pretrained language models for named entity recognition in historical texts. In *Proceedings of the 5th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 21–30, Punta Cana, Dominican Republic (online). Association for Computational Linguistics.

Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big?. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 610–623.

Steven Bethard, Guergana Savova, Wei-Te Chen, Leon Derczynski, James Pustejovsky, and Marc Verhagen. 2016. Semeval-2016 task 12: Clinical tempeval. In *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, pages 1052–1062.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. Legal-bert: The muppets straight out of law school. *arXiv preprint arXiv:2010.02559*.

A Conneau. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Agata Cybulska and Piek Vossen. 2014. Using a sledgehammer to crack a nut? lexical diversity and event coreference resolution. In *LREC*, pages 4545–4552.

Wietse De Vries, Andreas van Cranenburgh, Arianna Bisazza, Tommaso Caselli, Gertjan van Noord, and Malvina Nissim. 2019. Bertje: A dutch bert model. *arXiv preprint arXiv:1912.09582*.

Pieter Delobelle, Thomas Winters, and B. Berendt. 2020. Robbert: a dutch roberta-based language model. In *EMNLP*.

J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.

Jun Gao, Huan Zhao, Changlong Yu, and Ruifeng Xu. 2023. Exploring the feasibility of chatgpt for event extraction. *arXiv preprint arXiv:2303.03836*.

Ishaan Gulrajani and David Lopez-Paz. 2020. In search of lost domain generalization. *arXiv preprint arXiv:2007.01434*.

William L Hamilton, Jure Leskovec, and Dan Jurafsky. 2016. Cultural shift or linguistic drift? comparing two computational measures of semantic change. In *Proceedings of the conference on empirical methods in natural language processing. Conference on empirical methods in natural language processing*, volume 2016, page 2116. NIH Public Access.

Yu Hong, Wenxuan Zhou, Jingli Zhang, Guodong Zhou, and Qiaoming Zhu. 2018. Self-regulation: Employing a generative adversarial network to improve event detection. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 515–526.

Dieuwke Hupkes, Mario Giulianelli, Verna Dankers, Mikel Artetxe, Yanai Elazar, Tiago Pimentel, Christos Christodoulopoulos, Karim Lasri, Naomi Saphra, Arabella Sinclair, et al. 2022. State-of-the-art generalisation research in nlp: a taxonomy and review. *arXiv preprint arXiv:2210.03050*.

Dieuwke Hupkes, Mario Giulianelli, Verna Dankers, Mikel Artetxe, Yanai Elazar, Tiago Pimentel, Christos Christodoulopoulos, Karim Lasri, Naomi Saphra, Arabella Sinclair, et al. 2023. A taxonomy and review of generalization research in nlp. *Nature Machine Intelligence*, 5(10):1161–1174.

Heng Ji and Ralph Grishman. 2008. Refining event extraction through cross-document inference. In *Proceedings of ACL-08: Hlt*, pages 254–262.

Urja Khurana, Eric Nalisnick, and Antske Fokkens. 2021. How emotionally stable is albert? testing robustness with stochastic weight averaging on a sentiment analysis task. *arXiv preprint arXiv:2111.09612*.

Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, et al. 2021. Dynabench: Rethinking benchmarking in nlp. *arXiv preprint arXiv:2104.14337*.

Ross Deans Kristensen-McLachlan, Miceal Canavan, Márton Kardos, Mia Jacobsen, and Lene Aarøe. 2023. Chatbots are not reliable text annotators. *arXiv preprint arXiv:2311.05769*.

Andrey Kutuzov, Lilja Øvrelid, Terrence Szymanski, and Erik Velldal. 2018. Diachronic word embeddings and semantic shifts: a survey. *arXiv preprint arXiv:1806.03537*.

Anastasios Lamproudis, Aron Henriksson, and Hercules Dalianis. 2022. Evaluating pretraining strategies for clinical BERT models. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 410–416, Marseille, France. European Language Resources Association.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Qian Li, Jianxin Li, Jiawei Sheng, Shiyao Cui, Jia Wu, Yiming Hei, Hao Peng, Shu Guo, Lihong Wang, Amin Beheshti, et al. 2022. A survey on deep learning event extraction: Approaches and applications. *IEEE Transactions on Neural Networks and Learning Systems*.

Y. Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. 2019b. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692.

Enrique Manjavacas and Lauren Fonteyn. 2021. Macberth: Development and evaluation of a historically pre-trained language model for english (1450-1950). In *Proceedings of the Workshop on Natural Language Processing for Digital Humanities*, pages 23–36.

Enrique Manjavacas and Lauren Fonteyn. 2022a. Adapting vs. pre-training language models for historical languages. *Journal of Data Mining & Digital Humanities*, (Digital humanities in languages).

Enrique Manjavacas and Lauren Fonteyn. 2022b. Non-parametric word sense disambiguation for historical languages. In *Proceedings of the 2nd International Workshop on Natural Language Processing for Digital Humanities*, pages 123–134.

Martin Müller, Marcel Salathé, and Per E Kummervold. 2023. Covid-twitter-bert: A natural language processing model to analyse covid-19 content on twitter. *Frontiers in artificial intelligence*, 6:1023281.

Thien Nguyen and Ralph Grishman. 2018. Graph convolutional networks with argument-aware pooling for event detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Barbara Plank. 2016. What to do about non-standard (or non-canonical) language in nlp. *arXiv preprint arXiv:1608.07836*.

James Pustejovsky, Kiyong Lee, Harry Bunt, and Laurent Romary. 2010. Iso-timeml: An international standard for semantic annotation. In *LREC*, volume 10, pages 394–397.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of nlp models with checklist. *arXiv preprint arXiv:2005.04118*.

Roser Saurí, Jessica Littman, Bob Knippen, Robert Gaizauskas, Andrea Setzer, and James Pustejovsky. 2006. TimeML annotation guidelines version 1.2. 1.

Rachele Sprugnoli and Sara Tonelli. 2019. Novel event detection and classification for historical texts. *Computational Linguistics*, 45(2):229–265.

William F. Styler IV, Steven Bethard, Sean Finan, Martha Palmer, Sameer Pradhan, Piet C. De Groen, Brad Erickson, Timothy Miller, Chen Lin, Guergana Savova, et al. 2014. Temporal annotation in the clinical domain. *Transactions of the association for computational linguistics*, 2:143–154.

Wondimagegnhue Tsegaye Tufa, Ilia Markov, and Piek Vossen. 2024. Unknown script: Impact of script on cross-lingual transfer. *arXiv preprint arXiv:2404.18810*.

Naushad UzZaman, Hector Llorens, Leon Derczynski, James Allen, Marc Verhagen, and James Pustejovsky. 2013. Semeval-2013 task 1: Tempeval-3: Evaluating time expressions, events, and temporal relations. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 1–9.

Stella Verkijk, Pia Sommerauer, and Piek Vossen. 2024. Studying language variation considering the reusability of modern theories, tools and resources for annotating explicit and implicit events in centuries old text. In *Proceedings of the Eleventh Workshop on NLP for Similar Languages, Varieties, and Dialects (VarDial 2024)*, pages 174–187.

Stella Verkijk and Piek Vossen. 2021. Medroberta. nl: a language model for dutch electronic health records. In *Computational Linguistics in the Netherlands*, volume 11, pages 141–159.

Stella Verkijk and Piek Vossen. 2023. Sunken ships shan't sail: Ontology design for reconstructing events in the dutch east india company archives. In *CEUR Workshop Proceedings*, page 320. CEUR Workshop Proceedings.

Piek Vossen. 2018. Newsreader at semeval-2018 task 5: Counting events by reasoning over event-centric-knowledge-graphs. In *Proceedings of the 12th international workshop on semantic evaluation*, pages 660–666.

Christopher Walker et al. 2006. ACE 2005 multilingual training corpus LDC2006T06. *Philadelphia: Linguistic Data Consortium*.

Xiaozhi Wang, Ziqi Wang, Xu Han, Wangyi Jiang, Rong Han, Zhiyuan Liu, Juanzi Li, Peng Li, Yankai Lin, and Jie Zhou. 2020. Maven: A massive general domain event detection dataset. *arXiv preprint arXiv:2004.13590*.

Ziqi Wang, Xiaozhi Wang, Xu Han, Yankai Lin, Lei Hou, Zhiyuan Liu, Peng Li, Juanzi Li, and Jie Zhou. 2021. Cleve: contrastive pre-training for event extraction. *arXiv preprint arXiv:2105.14485*.

Meng Zhang, Zhiwen Xie, Jin Liu, Xiao Liu, Xiao Yu, and Bo Huang. 2024. Hypered: A hierarchy-aware network based on hyperbolic geometry for event detection. *Computational Intelligence*, 40(1):e12627.

Zhi Chang Zhang, Min Yu Zhang, Tong Zhou, and Yan Long Qiu. 2020. Pre-trained language model augmented adversarial training network for chinese clinical event detection. *Math. Biosci. Eng*, 17:2825–2841.

					GloBERTise	GloBERTise	BERTje	BERTje	mBERT	mBERT
inv_nr	year	#tokens	#g_ev	g_ev_dens	#pred_ev	f1	#pred_ev	f1	#pred_ev	f1
1066	1618	648	58	9%	34.0	0.61	33.4	0.58	37	0.52
1090	1626	3658	206	6%	178.0	0.46	172.8	0.39	187	0.41
1160	1647	2602	186	7%	127.8	0.53	95	0.36	104	0.40
1348	1679	281	32	12%	20.0	0.53	13	0.43	14	0.42
1430	1686	2088	184	9%	110.4	0.56	108	0.47	112	0.47
1439	1686	389	49	13%	26.0	0.58	17.2	0.44	32	0.60
1595	1697	2750	182	7%	145.0	0.57	90.4	0.45	135	0.52
8596	1707	1523	160	11%	135.2	0.72	114.4	0.62	121	0.64
4071	1713	489	62	12%	19.0	0.40	16.2	0.29	15	0.35
7673	1716	611	45	7%	59.4	0.62	44.6	0.47	56	0.53
9001	1720	3423	166	5%	190.8	0.58	128.2	0.38	171	0.49
11012	1736	3881	301	8%	164.0	0.46	237.8	0.42	237	0.48
2665	1746	242	21	9%	11.0	0.62	9.4	0.58	11	0.56
2693	1747	439	26	6%	17.2	0.55	12.4	0.41	17	0.53
3476	1777	2194	138	6%	142.2	0.59	96.4	0.39	150	0.52

Table 4: Amount of predicted events and F1 scores per fold of various models averaged over 5 runs. Information on the document used as test data in each fold is provided. #g_ev = number of gold event tokens; g_ev_dens = gold event density, i.e. percentage of tokens in the test set that refers to an event; #pred_ev: total number of (correctly and incorrectly) predicted events by the model.

	Parameters	Data (tokens / bytes)
BERT (base)	110M	3.3B / 16GB
RoBERTa (base)	125M	/ 160GB
MacBERTh	110M	3.9B

Table 5: Information on English models tested

	BERT	RoBERTa	MacBERTh	lex_baseline
P-event	0.24	0.55	0.38	0.83
R-event	0.06	0.25	0.06	0.22
F1-event	0.08	0.32	0.10	0.34
mention acc.	0.09	0.35	0.08	0.34

Table 6: Scores on detecting the event class averaged over data folds and seeds

	BERT	RoBERTa	MacBERTh
P-event	0.285	0.070	0.204
R-event	0.076	0.086	0.034
f1-event	0.107	0.082	0.054

Table 7: Standard deviation scores between seeds for each model

learning_rate	5e-05
per_device_train_batch_size	32
per_device_test_batch_size	32
num_train_epochs	5
weight_decay	0.01
seeds	[23052024, 21102024, 553311, 6834, 888]

Table 8: Parameter settings for finetuning