Triple E - Effective Ensembling of Embeddings and Language Models for NER of Historical German.

Stefan Schweter[1] and Luisa März[2]

[1] Bayerische Staatsbibliothek München, Digital Library/Munich Digitization Center
stefan.schweter@bsb-muenchen.de
[2] Center for Information and Language Processing (CIS), LMU Munich
maerz@cis.lmu.de

Abstract. Named entity recognition (NER) for historical texts is a challenging task compared to NER for contemporary texts. Historical texts come with several peculiarities that differ greatly from modern texts and large labeled corpora for training a neural tagger are hardly available. In this work we tackle NER for historical German with an ensembling approach, combining different labeled and unlabeled resources of historical and contemporary texts as part of the CLEF HIPE 2020 evaluation lab. We stack different word/subword embeddings and transformer-based language models to train a powerful NER tagger for historical German. We conduct experiments with different word embeddings, Flair embeddings and pretrained Bert models. The named entities are classified in literal and in metonymic sense, for which we have developed a separate tagger each. Our experiments show that the usage of Bert is particularly helpful, when trained on a large amount of historical data. Our best ensemble is a combination of FastText embeddings trained on German Wikipedia, Flair embeddings trained on CLEF HIPE data (historical German) and a Bert language model trained on a large corpus of historical German. We release our code and models[3].

Keywords: Named Entity Recognition · Transformer-based language models · Embeddings · Historical texts · Flair · FastText · Byte Pair Encoding.

1 Introduction

In NER neural networks achieve good accuracy on high resource domains such as modern news text or Twitter ([2, 4]). But on historical text, NER taggers often perform poorly. This is due to domain shift and to the fact that historical texts contain systematic errors not found in modern text, since historical datasets

[3] Our code and models are available at: https://github.com/stefan-it/clef-hipe

usually stem from optical character recognition (OCR). OCR is noisy and the Gothic type face (Fraktur) is a low resource font, that is very challenging for OCR. Another problem is that a large amount of data is required when training neural models and only relatively small corpora (e.g. [20]) exist for historical NER. All of these challenges mean that NER for contemporary texts differs greatly from NER for historical texts and that existing models cannot be used. From a resource orientated and ecological point of view it is reasonable to reuse existing models to save both computing power and emissions. Therefore, we reuse existing models on the one hand and make our newly developed language models publicly available on the other hand.

In the NLP community there are several approaches and models provided, one of which is Flair [1]. Flair allows to apply state-of-the-art natural language processing (NLP) models, such as NER, part-of-speech tagging (PoS), word sense disambiguation or classification to various input texts. In this work we built our systems with that framework.

Transformer-based language models are widely used and Bert [8] can be considered as a powerful standard resource. There are several recent approaches that use Bert for NER in different languages, such as [25] or [16]. The latter conduct experiments with historical German using Bert and unsupervised pretraining on a large corpus of historical German texts together with supervised pretraining on a contemporary German corpus.

1.1 Task and Objective

In this work, we address neural NER tagging on historical German data. With our approach we aim to solve coarse grained NER in the CLEF HIPE shared task [11] (bundle 4) for historical German as best as possible. The tagset of the provided data contains person, location, organisation, product and time. The organizers arranged two scenarios to be solved: NER for the literal sense of the words and NER for metonymic sense. The example below shows that the tags for the literal (first sentence) and metonymic (second sentence) sense can differ. *Hannover* can be interpreted as an organization as well as a location depending on its context and the metonymic category addresses this issue.

Example:
Unterhandlungen über das Konkordat mit ***B-loc Hannover*** *schreiten voran.*
Unterhandlungen über das Konkordat mit ***B-org Hannover*** *schreiten voran.*
(Negotiations on the Concordat with Hanover are progressing.)

This paper is structured as follows: The next section describes data sets and other resources that are used in the experiments presented. Section 3 outlines our method and section 4 explains details on implementation and the conducted experiments. The outcome of the experiments is discussed in that section as well. Then, section 5 overviews ideas for future work and we conclude the paper with section 6.

2 Data and Resources

This section describes the data provided by the shared task organizers as well as additional resources and data that we used for our experiments.

2.1 CLEF HIPE Data

The shared task corpus for German is composed of articles sampled among several Swiss and Luxembourgish historical newspapers on a diachronic basis and is provided by the CLEF-HIPE-2020 organizers. The articles that were chosen for the train, development and test data are journalistic articles only, that had to match certain selection criteria such as length or format. Feuilleton, tabular data, crosswords, weather forecasts, time schedules and obituaries were excluded as well as articles that were fully illegible due to massive ORC noise. The newspaper content stems for the time period from 1798 until 2018 and thus there is different OCR quality present in the data which covers a broad spectrum of text composition. The corpora were manually annotated by native speakers according the HIPE impresso guidelines ([10, 9]).

2.2 Additional Data and Resources

Table 1 gives an overview of all resources and shows time period and domain of each data set. The sizes of the training data used for the embeddings/models is shown in Table 2. Our approach includes data from different time periods as well as from various domains to reuse existing resources optimally.

Embeddings We use different **FastText**-based word embeddings [19] trained on Wikipedia[4], Common Crawl[5] and on historic data (provided by the organizers) as well as **Byte Pair Encoding**-based embeddings (BPE, [24]) trained on Wikipedia. We use the FastText embeddings trained on Wikipedia (*FastText Wiki*) and Common Crawl (*FastText CC*) in a "classic" word embeddings manner, that means we do not use subwords. To include subword information we use German subword embeddings [12] with a dimension of 300 and a vocab size of 200k (*BPEmb*). Additionally, we experiment with multilingual subword embeddings [13] with a dimension size of 300 and a vocab size of 1M (*MultiBPEmb*).

We use **Flair** embeddings [3, 2] provided by the organizers (*CLEF-HIPE*) and compared them to other Flair embeddings that were trained on historic data. We use two historic Flair embeddings that were trained by [23]: embeddings trained on the *Hamburger Anzeiger* newspaper corpus (*HHA*) and embeddings trained on the *Wiener Zeitung* newspaper corpus (*WZ*). Both embeddings are available in the Flair framework. In addition we use the data of the recently published REDEWIEDERGABE corpus [6] that consists of fictional and non-fictional texts. We also experiment with the Flair embeddings provided by [3] (*German* Flair).

[4] `https://fasttext.cc/docs/en/pretrained-vectors.html`
[5] `https://fasttext.cc/docs/en/crawl-vectors.html`

usage	name	time period	domain
train data		1798 - 2018	news
FastText	FastText Wiki	contemp.	various
FastText	FastText CC	contemp.	various
BPE	BPEmb	1798 - 2018	news
BPE	MultiBPEmb	contemp.	news
Flair	HHA	1888 - 1945	news
Flair	WZ	1703 - 1875	news
Flair	Redewiedergabe	1840 - 1920	various
Flair	*German* Flair	contemp.	various
Flair	CLEF-HIPE	1798 - 2018	news
Bert	*Europeana* Bert	1618 - 1990	news
Bert	*German* Bert	historical	various

Table 1. Overview of time periods and domains of the training data used for the embeddings and language models.

usage	name	data	tokens	size
train data		CLEF HIPE*	0.071	S
FastText	FastText Wiki	Wikipedia	1400	L
BPE	BPEmb	Wikipedia	$\approx$ 1400	L
BPE	MultiBPEmb	Wikipedia	$<$ 7000	L
FastText	FastText CC	Common Crawl	65648	XL
Flair	Redewiedergabe	REDEWIEDERGABE	0.489	S
Flair	*German* Flair	OPUS project	500	M
Flair	HHA	Hamburger Anzeiger	742	M
Flair	WZ	Wiener Zeitung	802	M
Flair	CLEF-HIPE	CLEF-HIPE*	1722	L
Bert	*Europeana* Bert	Europeana	8000	L
Bert	*German* Bert	-	$\approx$ 24000	XL

Table 2. Overview of different training data used. Number of tokens is given in millions. * indicates that data was provided by the organizers.

Transformer-based language models For transformer-based language models we conduct experiments with self-trained BERT models, *Europeana* BERT[6] and large German BERT[7] (*German* BERT). In preliminary experiments we also used publicly available German BERT models (deepset[8] and DBMDZ[9]). Since their performance was not convincing we did not include them in our final setup.

The *Europeana* BERT data comes from the Europeana Newspapers collection[10], which contains historical news articles in 12 languages published between 1618 and 1990. The *Europeana* BERT model was trained on 51GB of newspapers, extracted from German Europeana. It mainly covers newspaper articles from the 18th to 20th century. *German* BERT was trained on a huge collection of various historical resources.

3 Methods

To develop an efficient NER tagger for historical texts we experiment with stacking methods described in the following.

We experiment with different kinds of ensembling/stacking approaches on the development set to figure out the optimal combination of embeddings and language models. Our final system CISTERIA uses an ensemble of word embeddings, transformer-based language models and FLAIR embeddings. To arrive at the best combination of embeddings for CISTERIA we conduct experiments where we a) select the best word embeddings, FLAIR embeddings and transformer-based language models independently and b) combine the best selected word embedding, the best transformer-based language model and the best FLAIR embeddings and feed those to our network. The network for the classification is a bidirectional LSTM with a conditional random field (CRF) as final output layer as proposed by [14]. Note that we train separate models for the metonymic and the literal sense span.

4 Implementation and Experiments

The following describes the implementation of our approach, overviews the different experiments and presents the results. Our final system for the CLEF HIPE 2020 evaluation lab is referred to as CISTERIA.

To feed the CLEF-HIPE data into our tagger we need several preprocessing steps. Our preprocessing includes sentence splitting (rule based method) and normalizing word hyphenations. The motivation behind normalizing hyphenation is that pretrained language models normally include normalized text and the word hyphenation character in the CLEF-HIPE shared task is a special symbol (¬)

[6] https://github.com/stefan-it/europeana-bert
[7] Under review.
[8] https://huggingface.co/bert-base-german-cased
[9] https://github.com/dbmdz/berts
[10] http://www.europeana-newspapers.eu/

and does not occur in training corpora for pretrained language models. As we use contextualized word embeddings, the correct hyphenation is very important to produce high quality embeddings. To get the data ready for evaluation with the officially provided evaluation script, we perform a reverse process and add word hyphenation and sentence boundaries again.

We use the Flair [1] library to train our NER tagging models and we make use of Bert embeddings in a *feature-based* setting. In order to get a representation for an input token, we first compute the mean of the first subword over all layers of the transformer-based architecture and feed the resulting representation into a bidirectional LSTM with a CRF as the final layer, following [3]. To ensemble different embeddings and language models their representations are concatenated and the resulting vector is processed by the neural model. Cisteria was trained on the official training and development data and does not use any other additional labeled training data.

For the experiments with transformer-based language models, we fine-tune Bert models using the Hugging Face Transformers library [29]. For these fine-tuning experiments, we use a batch size of 16 and train 10 epochs. We perform three runs per transformer-based model and select the best model based on development F1-score. We do not perform extensive hyperparameter search.

We then use the fine-tuned model in Flair (feature-based approach) for all further experiments. We use a bidirectional LSTM with 256 hidden states and a batch size of 16. The original Bert paper [8] uses the last four layers of the transformer-based model for a feature-based NER model. Additionally, we reduce the learning rate by a factor of 0.5 with a patience of 3. This factor determines the number of epochs with no improvement after which the learning rate will be reduced and can be seen as early stopping.

We found that fine-tuning a Bert model for the metonymic sense span was very unstable resulting in zero F1-scores. This is a well known problem for datasets when only a small number of training instances are available and a solution could be to use a different dropout strategy [17]. For that reason we trained a model using the CLEF-HIPE Flair embeddings. In the prediction phase we only do predictions when an entity is detected for the literal sense span.

Our final system for the literal sense span uses FastText embeddings trained on Wikipedia (*FastText Wiki*) and a self-trained large *German* Bert model. For the metonymic sense span we train a separate model that uses FastText embeddings trained on Wikipedia and Flair embeddings provided by the organizers.

4.1 Results

For the evaluation of NER there are two regimes: strict and fuzzy. The strict regime corresponds to exact boundary matching whereas the fuzzy takes overlapping boundaries into account, a detailed description can be found in [11]. In

addition spans are evaluated w.r.t literal or metonymic sense (see section 1.1). We evaluate our systems using the official evaluation script[11].

All our reported results on the development set refer to the F1 score for coarse grained NER in the strict scenario for the literal sense. For the test set we report precision, recall and F1 score for both scenarios in the literal sense as well as in the metonymic sense (see Table 8). According to the overview paper of the shared task [11] the baseline in the strict evaluation scenario for German Coarse NER in literal sense results in **47.6%** F1-score (see Table 7).

Our results of the experiments with different word embeddings show that the *FastText Wiki* embeddings perform best, see Table 3. With an F1-score of approx. 69% they can overcome the baseline by more than 20 percentage points. Interesting is that the *FastText Wiki* embeddings are not trained on the biggest amount of data compared to the other word embeddings (see Table 2).

Model	F1
FastText Wiki	**69.28** ±0.65
FastText CC	66.38 ±0.51
BPEmb [12]	67.71 ±0.48
MultiBPEmb [13]	66.22 ±0.14

Table 3. Experiments with different word Embeddings on German development set. Averaged F1-score over 3 runs is reported here. Best result in bold.

Different Flair embeddings lead consistently to better results than using word embeddings. The Flair embeddings provided by the organizers (*CLEF-HIPE*) perform best, with an F1-score of 77.04% (see Table 4). The gap between the different Flair embeddings is comparably large and ranges from seven to three percentage points difference. Here the embeddings that were trained on the biggest amount of data perform best and the *Redewiedergabe* embeddings that were trained on the least amount perform worst.

Model	F1
Hamburger Anzeiger [23]	74.14 ±0.11
Wiener Zeitung [23]	75.07 ±0.11
Redewiedergabe [6]	70.21 ±0.27
German (Flair) [3]	74.98 ±0.30
CLEF-HIPE	**77.04** ±0.12

Table 4. Experiments with different Flair Embeddings on German development set. Averaged F1-score over 3 runs is reported here. Best result in bold.

[11] `https://github.com/impresso/CLEF-HIPE-2020-scorer`

Model	F1
Europeana Bert (cased)	80.41 ±0.14
Europeana Bert (uncased)	79.66 ±0.32
German Bert (cased, large)	**82.11** ±0.50

Table 5. Experiments with different Bert models on German development set. Averaged F1-score over 3 runs is reported here. Best result in bold.

The usage of Bert enhances the performance once more. The *German* Bert model performs best and results in 82.11% F-score (see Table 5). Again this is the model that was trained on the biggest amount of data. The cased version of *Europeana* Bert leads to a similar performance with approx. two percentage points less. Since German is case sensitive it is understandable that the cased models perform better than the uncased ones. Like with the Flair embeddings every setup with Bert outperforms the models of our previous experiments.

Model	F1
FastText (Wikipedia) + CLEF-HIPE + *German* Bert	83.57 ±0.36
FastText (Wikipedia) + CLEF-HIPE	77.97 ±0.47
FastText (Wikipedia) + *German* Bert	**83.69** ±0.08

Table 6. Experiments with different stacking experiments on German development set. Averaged F1-score over 3 runs is reported here. Best result in bold.

Finally the combination of *German* Bert with the *FastText Wiki* embeddings outperforms all of our other systems on the development set and results in 83.69% (see Table 6). This result is plausible if we compare it to the best F1-scores of [16] on other historical datasets. For two datasets their performance is around 84%. The addition of the best Flair embeddings decreases the results slightly. If combining the best Flair embeddings with the best FastText embeddings the model performs better than using Flair embeddings only but still worse than the other stacking approaches. The performance of our best system is approx. 40% better than the baseline, which is a large improvement.

4.2 Discussion of Results

We want to relate our final results on the test set to those of the other participating teams. Compared to the baseline our final systems (CISTERIA) could perform very good. If we take a look at the median of all participating teams our system for the literal sense performs approx. 2% points better in the strict scenario and is almost on par with the median in the fuzzy scenario (see Table 7). For both regimes the best system *L3i* [5] outperforms ours by slightly more than

10% points. This could be due to the fact that they use powerful transformer-based embeddings for different languages and a hierarchical transformer-based attention model [28] together with a multi task learning setting approach. Our experiments with BERT embeddings show that the model can benefit from the German Europeana BERT language model a lot and that only a model trained with even more data could outperform it. Therefore it is not surprising that a model trained with more of these powerful BERT embeddings performs even better. The benefit of the combination of models for different languages is at hand and we suppose that our model performances can be enhanced if we integrate multilinguality as well.

Team	Strict			Fuzzy		
	P	R	F1	P	R	F1
CISTERIA	0.745	0.578	0.651	**0.880**	0.683	0.769
EHRMAMA [27]	0.697	0.659	0.678	0.814	0.765	0.789
L3I [5]	**0.790**	**0.805**	**0.797**	0.870	**0.886**	**0.878**
SBB [15]	0.499	0.484	0.491	0.730	0.708	0.719
SINNER [21]	0.658	0.658	0.658	0.775	0.819	0.796
UPB [7]	0.677	0.575	0.621	0.788	0.740	0.763
UVA-ILPS [22]	0.499	0.556	0.526	0.689	0.768	0.726
WEBIS [26]	0.695	0.337	0.454	0.833	0.405	0.545
Baseline	0.643	0.378	0.476	0.790	0.464	0.558
Median	0.686	0.576	0.636	0.801	0.752	0.766

Table 7. Results for NERC-Coarse literal with micro precision, recall and F1-score on the test set. Bold font indicates highest, underlined the second highest result.

In the evaluation w.r.t the metonymic sense it turns out that our approach to train a separate model was constructive. In both regimes our system performs clearly above the median and in the fuzzy regime our F1-score is the second best (see Table 8). Again the *L3i* system can reach the best scores, probably due to the same reasons as mentioned above. Our results support our strategy that we only do predictions for tokens where the literal sense is classified as an entity.

Regarding the precision our system performs very well and reaches second best performance in all cases, except for the fuzzy evaluation in the literal sense where our system performs best. Unfortunately the recall is relatively low with around 50% for the metonymic sense and 57%/68% for the strict/fuzzy evaluation in the literal sense. Our system has the ability to classify correctly if it identifies a token as a possible entity but has problems with finding the entities as such.

Team	Strict			Fuzzy		
	P	R	F1	P	R	F1
Cisteria	0.738	0.500	0.596	0.787	0.534	0.636
Ehrmama [27]	0.696	0.542	0.610	0.707	0.551	0.619
L3i [5]	0.571	**0.712**	**0.634**	0.626	**0.780**	**0.694**
Baseline	**0.814**	0.297	0.435	**0.814**	0.297	0.435

Table 8. Results for NERC-Coarse metonymic with micro precision, recall and F1-score. Bold font indicates highest, underlined the second highest result.

5 Future Work

The approach of the winning team suggests to include multilingual language models and/or more data. Since a lot of powerful pretrained language models are available we will integrate some of them in Cisteria.

Another strategy is to take into account the domain of historical language even more. Since there is a lot of noise in the data due to OCR it greatly differs from modern standard language. Nevertheless there are many modern corpora available on which transformer-based language models can be trained. Our goal is to increase the similarity of those modern corpora to historical data. Therefore we want to recreate some of the phenomena in historical corpora in the modern corpora that we use for training the language models.

Besides that, manual rule-based sentence segmentation could have drawbacks (e.g. bad segmentation could lead to short sentences). So in future experiments we could use the context before and after the actual training sentence, such as in [18]. This approach could eliminate potential drawbacks of an automatically sentence segmented training corpus, because shorter sentences are now enhanced with longer contexts.

6 Conclusion

We proposed a system to solve coarse grained NER for German in the CLEF HIPE shared task. We conducted experiments with ensembling different word and subword embeddings as well as transformer-based language models on the basis of a bidirectional LSTM with a CRF as final layer. To use historical resources at best we trained large language models on historical German data, such as the German Europeana collection. Our best system uses FastText embeddings trained on German Wikipedia data in combination with a large German Bert language model. With a performance of 65.1% F1-score our best system performs slightly better than the median in the strict scenario for the literal sense and with an F1-score of 76.9% on par with the median in the fuzzy scenario. For the metonymic sense our best system performs clearly above the baseline and reaches the second best performance in the fuzzy scenario.

References

1. Akbik, A., Bergmann, T., Blythe, D., Rasul, K., Schweter, S., Vollgraf, R.: 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). pp. 54–59. Association for Computational Linguistics, Minneapolis, Minnesota (Jun 2019), https://www.aclweb.org/anthology/N19-4010
2. Akbik, A., Bergmann, T., Vollgraf, R.: Pooled Contextualized Embeddings for Named Entity Recognition. 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). pp. 724–728. Association for Computational Linguistics, Minneapolis, Minnesota (Jun 2019), https://www.aclweb.org/anthology/N19-1078
3. Akbik, A., Blythe, D., Vollgraf, R.: Contextual String Embeddings for Sequence Labeling. In: Proceedings of the 27th International Conference on Computational Linguistics. pp. 1638–1649. Association for Computational Linguistics, Santa Fe, New Mexico, USA (Aug 2018), https://www.aclweb.org/anthology/C18-1139
4. Baevski, A., Edunov, S., Liu, Y., Zettlemoyer, L., Auli, M.: Cloze-driven Pretraining of Self-attention 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). Association for Computational Linguistics, Hong Kong, China (Nov 2019), https://www.aclweb.org/anthology/D19-1539
5. Boros, E., Linhares Pontes, E., Cabrera-Diego, L.A., Hamdi, A., Moreno, J.G., Sidère, N., Doucet, A.: Robust Named Entity Recognition and Linking on Historical Multilingual Documents. In: Cappellato, L., Eickhoff, C., Ferro, N., Névéol, A. (eds.) CLEF 2020 Working Notes. Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum. CEUR-WS (2020)
6. Brunner, A., Engelberg, S., Jannidis, F., Tu, N.D.T., Weimer, L.: Corpus REDEWIEDERGABE. In: Proceedings of The 12th Language Resources and Evaluation Conference. pp. 803–812. European Language Resources Association, Marseille, France (May 2020), https://www.aclweb.org/anthology/2020.lrec-1.100
7. Craita, C.C., Cercel, D.C.: Multilingual Named Entity Recognition on Historical Texts Using Transfer and Multi-Task Learning. In: Cappellato, L., Eickhoff, C., Ferro, N., Névé ol, A. (eds.) CLEF 2020 Working Notes. Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum. CEUR-WS (2020)
8. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: 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). pp. 4171–4186. Association for Computational Linguistics, Minneapolis, Minnesota (Jun 2019), https://www.aclweb.org/anthology/N19-1423
9. Ehrmann, M., Romanello, M., Flückiger, A., Clematide, S.: HIPE - Shared Task Participation Guidelines (v1.1) (2020). https://doi.org/10.5281/zenodo.3677171
10. Ehrmann, M., Romanello, M., Flückiger, A., Clematide, S.: Impresso Named Entity Annotation Guidelines (Jan 2020). https://doi.org/10.5281/zenodo.3604227
11. Ehrmann, M., Romanello, M., Flückiger, A., Clematide, S.: Overview of CLEF HIPE 2020: Named Entity Recognition and Linking on Historical Newspapers. In:

Arampatzis, A., Kanoulas, E., Tsikrika, T., Vrochidis, S., Joho, H., Lioma, C., Eickhoff, C., Névéol, A., Cappellato, L., Ferro, N. (eds.) Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the 11th International Conference of the CLEF Association (CLEF 2020). Lecture Notes in Computer Science (LNCS), vol. 12260. Springer (2020)

12. Heinzerling, B., Strube, M.: BPEmb: Tokenization-free Pre-trained Subword Embeddings in 275 Languages. In: chair), N.C.C., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Hasida, K., Isahara, H., Maegaard, B., Mariani, J., Mazo, H., Moreno, A., Odijk, J., Piperidis, S., Tokunaga, T. (eds.) Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018). European Language Resources Association (ELRA), Miyazaki, Japan (May 7-12, 2018 2018)
13. Heinzerling, B., Strube, M.: Sequence Tagging with Contextual and Non-Contextual Subword Representations: A Multilingual Evaluation. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 273–291. Association for Computational Linguistics, Florence, Italy (Jul 2019), https://www.aclweb.org/anthology/P19-1027
14. Huang, Z., Xu, W., Yu, K.: Bidirectional LSTM-CRF models for sequence tagging. arXiv preprint arXiv:1508.01991 (2015)
15. Labusch, K., Neudecker, C.: Named Entity Disambiguation and Linking Historic Newspaper OCR with BERT. In: Cappellato, L., Eickhoff, C., Ferro, N., Névéol, A. (eds.) CLEF 2020 Working Notes. Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum. CEUR-WS (2020)
16. Labusch, K., Neudecker, C., Zellhöfer, D.: Bert for named entity recognition in contemporary and historic german. In: Proceedings of the 15th Conference on Natural Language Processing (KONVENS 2019): Long Papers. pp. 1–9. German Society for Computational Linguistics & Language Technology, Erlangen, Germany (2019)
17. Lee, C., Cho, K., Kang, W.: Mixout: Effective Regularization to Finetune Large-scale Pretrained Language Models. In: International Conference on Learning Representations (2020), https://openreview.net/forum?id=HkgaETNtDB
18. Luoma, J., Pyysalo, S.: Exploring Cross-sentence Contexts for Named Entity Recognition with BERT. arXiv e-prints arXiv:2006.01563 (Jun 2020)
19. Mikolov, T., Grave, E., Bojanowski, P., Puhrsch, C., Joulin, A.: Advances in Pre-Training Distributed Word Representations. In: Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018) (2018)
20. Neudecker, C.: An Open Corpus for Named Entity Recognition in Historic Newspapers. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16). pp. 4348–4352. European Language Resources Association (ELRA), Portorož, Slovenia (May 2016), https://www.aclweb.org/anthology/L16-1689
21. Ortiz, S., Pedro, J., Dupont, Y., Lejeune, G., Tian, T.: SinNer@Clef-Hipe2020: Sinful adaptation of SotA models for Named Entity Recognition in historical French and German newspapers. In: Cappellato, L., Eickhoff, C., Ferro, N., Névé ol, A. (eds.) CLEF 2020 Working Notes. Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum. CEUR-WS (2020)
22. Provatorova, V., Vakulenko, S., Kanoulas, E., Dercksen, K., van Hulst, J.M.: CLEF HIPE Working Notes: UvA ILPS & REL. In: Cappellato, L., Eickhoff, C., Ferro, N., Névé ol, A. (eds.) CLEF 2020 Working Notes. Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum. CEUR-WS (2020)

23. Schweter, S., Baiter, J.: Towards Robust Named Entity Recognition for Historic German. In: Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019). pp. 96–103. Association for Computational Linguistics, Florence, Italy (Aug 2019), https://www.aclweb.org/anthology/W19-4312
24. Sennrich, R., Haddow, B., Birch, A.: 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). pp. 1715–1725. Association for Computational Linguistics, Berlin, Germany (Aug 2016), https://www.aclweb.org/anthology/P16-1162
25. Souza, F., Nogueira, R., Lotufo, R.: Portuguese Named Entity Recognition using BERT-CRF (2019)
26. Tobollik, T., Wiegmann, M., Wolska, M., Stein, B.: Enrichement-based Oversampling for Coarse-grained NER in Historical Text. In: Cappellato, L., Eickhoff, C., Ferro, N., Névé ol, A. (eds.) CLEF 2020 Working Notes. Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum. CEUR-WS (2020)
27. Todorov, K., Colavizza, G.: Transfer Learning for Named Entity Recognition in Historical Corpora. In: Cappellato, L., Eickhoff, C., Ferro, N., Névéol, A. (eds.) CLEF 2020 Working Notes. Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum. CEUR-WS (2020)
28. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention Is All You Need. CoRR **abs/1706.03762** (2017), http://arxiv.org/abs/1706.03762
29. Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., Drame, M., Lhoest, Q., Rush, A.M.: HuggingFace's Transformers: State-of-the-art Natural Language Processing. arXiv e-prints arXiv:1910.03771 (Oct 2019)