Exploring Transformers for Multilingual Historical Named Entity Recognition*

Anja Ryser[1,†], Quynh Anh Nguyen[1,2,†], Niclas Bodenmann[1,†] and Shih-Yun Chen[1,3,†]

[1]University of Zurich, Rämistrasse 21, 8006 Zürich, Switzerland
[2]University of Milan, Via Festa del Perdono, 7, 20122 Milano MI, Italy
[3]Zurich University of Applied Sciences, Gertrudstrasse 15, 8401 Winterthur, Switzerland

Abstract

This paper explores the performance of out-of-the-box transformers language models for historical Named Entity Recognition (NER). Within the HIPE2022 (Identifying **Hi**storical **P**eople, **P**laces, and other **E**ntities) shared task, we participated in the NER-COARSE task of the Multilingual Newspaper Challenge (MNC). Three main approaches are experimented with: ensembling techniques on multiple fine-tuned models, using multilingual pretrained models, and relabeling the entity tags from the IOB-segmentation to a simplified version. By ensembling predictions from different system outputs, we outperformed the baseline model in the majority of cases. Moreover, through post-submission experiments, we found that using multilingual models did not yield better results compared to monolingual models. Furthermore, the relabeling experiment on the Newseye French dataset showed that merging entity labels and inferring the IOB segmentation in postprocessing increases precision but lowers recall. Last but not least, soft-label ensembling experiments on the same dataset enhanced precision, recall and thus F1-scores compared to hard-label ensembling by at least one percentage point.

Keywords

Named Entity Recognition, Historical Newspaper, HIPE2022, Transfer Learning, Transformers, Multilingual Models

1. Introduction

Named Entity Recognition (NER) on historical newspaper text is a task with many pitfalls. Differences in language, its use, and the world it refers to, as well as technical artifacts, make models that perform well in contemporary texts significantly worse in historical texts. With our contribution to the HIPE2022 Shared Task, we explore the performance of transformers-architectures pretrained on historical and contemporary data available via HuggingFace [1]. We combine these models with task-specific knowledge in pre- and postprocessing, and in post-submission experiments, we further investigate the performance of only predicting on categories (without using IOB encoding), soft-label ensembling, and multilingual language models.

CLEF 2022: Conference and Labs of the Evaluation Forum September 05–08, 2022, Bologna, Italy

*Named Entity Recognition in historical newspapers

†These authors contributed equally.

anja.ryser@uzh.ch (A. Ryser); quynhanh.nguyen@uzh.ch (Q. A. Nguyen); niclaslinus.bodenmann@uzh.ch (N. Bodenmann); shih-yun.chen@uzh.ch (S. Chen)

© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

CEUR-WS.org/Vol-3180/paper-86.pdf

2. Related Work

Transformers [2] has rapidly become the dominant architecture for natural language processing, surpassing alternative neural models such as convolutional and recurrent neural networks in performance for tasks in both natural language understanding and natural language generation [3]. The Transformers architecture is particularly conducive to pretrain on large text corpora, leading to major gains in accuracy on downstream tasks[3]. As a result, the release of pretrained contextualised word embeddings such as BERT [4] pushed further the upper bound of modern NER performances [5] and established state-of-the-art results for modern NER [5] [6].

In the HIPE2020 Shared Task, several top solutions were developed based on pretrained language model embeddings with transformers-based architectures. Ghannay et al. [7] achieved the second-best result for French with an 81% F1-score in the strict scenario by using CamemBERT [8], a multi-layer bidirectional transformer similar to RoBERTa [9], together with a CRF decoder. Todorov et al. [10] implemented an architecture made of a modular embedding layer which was combined by newly trained and pre-trained embeddings, and a task-specific Bi- LSTM-CRF layer to handle NERC on coarse and fine-grained tags. They conclude that character-level embeddings, BERT, and a document-level data split are the most important factors in improving NER results. Besides, the experiment also shows that pretrained language models can be beneficial for NERC on low-resource historical corpora. Provatorova et al. [11] fine-tuned two pretrained BERT models [12], including *bert-base-cased* for English and *bert-base-multilingual-cased* for French and German. In order to enhance the robustness of the approach, a majority voting ensemble of 5 fine-tuned model instances was implemented per language. Their models achieved F1-scores of 68%, 52% and 47% for French, German and English respectively.

Section 4.2 describes how we employed multiple fine-tuned models, exploited ensembling techniques, and applied relabeling entities method.

3. Task and Datasets

We worked on coarse NER in digitized historical newspapers across different label sets and languages. More detailed information on this task can be found on the HIPE2022 website.

NER on historical newspapers poses its own unique challenges; non-standard language with old lexicon and syntax, errors from digitization such as errors in layout and optical character recognition (OCR) and the lack of resources for training make this task challenging [5].

We used a part of the data provided by the organizers of this task, namely 5 datasets of historical newspapers in English, German, French, Swedish and Finnish spanning from the 18th to the 20th century. The data contains newspapers digitized through different European cultural heritage projects. While most of the data were published before HIPE2022, some unpublished parts of the datasets were used as test-sets. Each dataset is annotated following different annotation guidelines and contains NER-tags and NEL-links to Wikidata. All datasets were provided in the HIPE-format [13]. Table 1 presents an overview of the historical newspaper datasets of HIPE2022 used in our experiments.

Resources We train our models on Google Colab with GPU enabled.

Table 1
Description of datasets contained in the HIPE2022-data

dataset	languages	comments
HIPE2020	de, en, fr	19-20C
Newseye	de, fi, fr, sv	19-20 C
Topres19th	en	19C, only location types
Sonar	de	19-20C
Letemps	fr	19-20C, unpublished

4. Methods

4.1. Data Preprocessing

We use a simple approach to preprocess the data. Lines with erroneous characters, empty lines, and lines containing metadata were removed while reading the tabulator-separated values (TSV) files. 'Nan'-values were filled with empty strings to keep the data structure intact.

Tokens are split into sentences using the `EndOfSentence`-tag provided in the data. The data is tokenized using the corresponding transformers model's tokenizer without any additional fine-tuning on our data.

4.2. Training

Models We employed a variety of models and pretrained weights for all different datasets. We distinguish between models that have been pretrained on historical data (historical language models, HLM) and on contemporary data. The HLMs mainly come from a single source, the Bavarian State Library [14]. We expect that the HLMs have already learned to deal with errors stemming from OCR, as these errors are prevalent in most historical datasets.

We mainly use BERT-based models [12, 15] but also experimented with XLNet [16] and ELECTRA [17]. See Table 7 for a full description of which pretrained models have been used. In the submitted run 1, all models listed were used for the ensembling. In run 2, the results of the single best model were submitted (marked in bold in the table). The models are instantiated with standard token classification heads from HuggingFace. [18, p. 98].

Training Parameters We fine-tune the models with the parameters coming from the pretrained models; were not these set, the default values of HuggingFace `TrainingArguments` have been used. All sentences were padded or truncated to a maximum length of 100 tokens. Because of the number of models we set out to deploy, we do not run a hyperparameter search. An initial experiment with label weights did not improve per- performance, and we returned to the defaults. We trained all models for three epochs.

4.3. Evaluation metrics

The evaluation metrics used for NER tasks in HIPE2022 are Precision, Recall, and F1 score on *macro* and *micro* levels. The same evaluation metrics are used to assess our systems. *F1-*

macro scores are computed on the document level and *F1-micro* scores on the entity-type level. Precisely, macro measures the average of the corresponding micro scores across all the documents, accounting for variance in document length but not for class imbalances.

Additionally, the model's performance was also measured in *strict* and *fuzzy*. In the strict scenario, a mention was only counted as correct when the exact gold-standard boundaries were met, whereas in the fuzzy evaluation, only a part of the mention needed to be recognized correctly. Because of this, in the *strict* measurement, predicting wrong boundaries leads to severe punishment, i.e., a mention is recognized, but one boundary is set wrong, leading to the whole entity being counted as False [13].

4.4. Inference

Single Models The single models we employ are initialized with a token classification head provided by HuggingFace. This is a linear mapping from the last encoder state to the output layer, where for every token, there are as many logits as labels in the dataset.

Because the gold labels are on whole words, while the models operate on subwords, we need a non-trivial mapping regime. In preprocessing, the label of the whole word is propagated down to all subwords. All subword logits belonging to a single word are summed up during inference, and the label with the highest score is chosen for the whole word. Ács et al. [19] evaluate different pooling strategies for subword aggregation. While they tend towards neural solutions such as an additional LSTM over the subword logits, they mention how the pooling strategy has a lower influence on NER (as opposed to morphological tasks such as POS- Tagging). Still, more advanced subword pooling strategies remain to be explored.

Ensembling On one dataset, predictions of all models were gathered, and the final label was chosen through a hard-ensembling method. The final prediction was the label with the most votes. In a tie between different labels, entity labels were favored, and between different entity labels, the choice was randomized.

Postprocessing We employed only one postprocessing rule for the shared task submission: If a token gets a label prediction starting with an `I` (inside) but is not preceded by an `I` or a `B` (beginning), it is changed to a `B`. Erroneously, we did not consider the label class. This was remedied in the post-submission experiments.

5. Post-Submission Experiments

This section introduces the three approaches we experimented with after the submission. We focused on Newseye French for the monolingual and all Newseye languages for multilingual approaches.

Motivation We tried to improve the submitted results. For better comparability of the different approaches of the post-submission experiments and due to time constraints, we decided to focus on one dataset and one language for the monolingual experiments. We saw the most potential

for improvement in the Newseye French dataset. Therefore, we used all available languages in the Newseye dataset for the multilingual approach. Our goal was to beat the baseline provided by the task organizers.

5.1. Soft-Label Ensembling

For the submission, we employed hard-label ensembling ('voting'). In this post-submission experiment, we evaluate the performance of soft-label ensembling on the Newseye French dataset. To infer the final label for a token, we average the probabilities (softmax logits) for the whole tokens of the individual models.

We follow Ju et al. [19], who argue for averaging the softmaxed logits because different models' logits might differ in magnitude. This is expected in our case, as the models use their own subword tokenization and, therefore, might sum over a different amount of subwords for the logits of the whole word.

The same models are used for the submission, presented in Table 7.

5.2. Multilingual Models

As shown in previous work, the performance of NLP- tasks can benefit from leveraging cross-lingual transfer learning and using multilingual models, which leads to more training data for a single model [20].

To test this, we used the same methods as described in chapter 4 with different multilingual BERT models (for further details, see Table 7) and Newseye data in all four available languages. In addition, we tested the single best model and the hard-label ensemble as described in paragraph 'Ensembling' in section 4.4.

The first trained model received the input sorted after language, which we assume could lead to catastrophic forgetting of the languages first seen. To avoid this, the sentences were shuffled before being fed in batches to the model during fine-tuning.

5.3. Relabeling

Through the error analysis in section 7.2, we noticed that one of the most occurring errors is **Right classification and wrong segmentation**, i.e., the model predicts *I-LOC* while the ground truth is *B-LOC*. We assume this is because the models are fine-tuned on the whole label set where B-tags and I-tags are handled as two different labels.

We chose the Newseye dataset in French to train the model used for this approach. All nine classes of the dataset, *['O', 'B-ORG', 'I-ORG', 'B-LOC', 'I-LOC', 'B-PER', 'I-PER', 'B-HumanProd', 'I-HumanProd']* are relabeled into five entity labels which are *['O', 'ORG', 'LOC', 'PER', 'HumanProd']*. After that, the text and corresponding new label of each word are used as the input of the training pipeline. The pipeline results in predictions with new labels that were reencoded. The IOB-tagging is reconstructed in the postprocessing. Table 13 shows more details of the results.

Table 2
F1-scores of Micro-strict evaluation of submitted ensembling system (Run 1)

dataset	en	fr	de	sv	fi	avg.
HIPE2020	0.513	0.678	**0.725**	-	-	**0.639**
Newseye	-	0.648	0.395	**0.643**	**0.567**	0.563
Topres19th	**0.787**	-	-	-	-	**0.787**
Sonar	-	-	0.490	-	-	**0.49**
Letemps	-	0.644	-	-	-	**0.644**

Table 3
F1-scores of Micro-strict evaluation of submitted best single model (Run 2)

dataset	en	fr	de	sv	fi	avg.
HIPE2020	x	**0.696**	0.695	-	-	0.696
Newseye	-	0.656	0.408	0.636	0.556	**0.564**
Topres19th	0.781	-	-	-	-	0.781
Sonar	-	-	0.477	-	-	0.477
Letemps	-	0.622	-	-	-	0.622

6. Results

To keep the results section concise, we focused on the micro-strict F1 score. From all measurements provided by the task organizers, micro-strict is the most punishing, resulting in lower scores. For more detailed results, see Tables 8 to 13 in the Appendix.

6.1. Submission

Table 2 and 3 show the F1-score over all labels for both submitted systems. 'avg' shows the average overall languages in each dataset. The best run for each language and averaged over all available languages between the two runs are marked in bold.

6.2. Post-Submission Experiments

Soft-Label Ensembling All scores benefit from switching from hard-label to soft-label ensembling by at least one percentage point from an average F1 score of 0.7 to 0.8 (see Table 4). However, our best run on the test set from the submission was not the (hard-label) ensembled model but the single model with the best scores on the validation set. With regards to Micro-F1 strict and fuzzy, the soft-label ensembled model is on par with the best individual model.

Multilingual models For this section, the test sets of the different languages were labeled and evaluated separately. The results shown in Table 5 are then averaged over all languages.

'submission ensembling' and 'submission best model' contain the results we handed in for submission, trained and ensembled monolingually and averaged over all languages. 'best model multilingual' is the best out of the five models used for ensembling. 'ensemble multilingual'

Table 4
Scores for all labels on Newseye French. 'Best Model (Run 2)' are the predictions of the best individual model with the best validation scores. The other two rows are different ensembling strategies over all models.

	Micro-P		Micro-R		Micro-F1		Macro-P		Macro-R		Macro-F1	
	strict	fuzzy	strict	fuzzy	strict	fuzzy	strict	fuzzy	strict	fuzzy	strict	fuzzy
Hard-Label (Run 1)	0.673	0.801	0.625	0.744	0.648	0.772	0.659	0.814	0.614	0.762	0.630	0.779
Best Model (Run 2)	0.655	0.785	**0.657**	**0.787**	0.656	0.786	0.630	0.775	0.623	**0.777**	0.621	0.766
Soft-Label	**0.685**	**0.818**	0.636	0.758	**0.659**	**0.787**	**0.677**	**0.829**	**0.63**	0.771	**0.649**	**0.793**

is a multilingually trained system with five different results which were then run through the ensembling process.

Table 5
Micro-strict scores averaged over all newseye testsets (de, fr, fi, sv) for experiments with multilingual models

System	Precision	Recall	F1
submission ensembling	0.70	0.65	0.67
submission best model	0.54	0.52	0.56
best model multilingual	0.62	0.55	0.58
ensemble multilingual	0.63	0.53	0.57

Results of the two runs of our submission show that ensembling improved the performance over all languages and performed better than the single models. In the multilingual experiments, the best model performs slightly better than the ensembling and beats the monolingual best model. Overall, the monolingual ensembling yielded the best results. We assume the multilingual ensembling results could be improved by excluding or replacing the worst performing model used in ensembling. Table 12 in the Appendix shows more detailed results.

Relabeling Table 13 shows the comparison between applying and not applying the relabeling method. The relabeling approach generally improves *precision* scores by around 1 to 2 percentage points.

The result shows notable changes in models performances regarding *precision* and *recall* scores. While Precision improves, *recall* scores slightly decrease compared to the performance of model without relabeling. As a consequence, F1-scores remain similar in both conditions.

Besides, the relabeling approach has also contributed to the marginal enhancement of model 4, i.e., the pretrained and fine-tuned French Europeana ELECTRA model. What stands out in Table 13 is the difference between model 4 using the relabeling method and model 4 not using relabeling method. All considered metrics uniformly rise around 0.3-3% with relabeling.

Table 6
Comparison of Micro-F1 for HIPE2020 German between the BERT-base LM trained on historical data and the BERT-base model LM on contemporary data.

	ALL		LOC		ORG		PERS		PROD		TIME	
	strict	fuzzy	strict	fuzzy	strict	fuzzy	strict	fuzzy	strict	fuzzy	strict	fuzzy
Historical	**0.702**	**0.805**	**0.814**	**0.870**	0.441	**0.572**	**0.690**	**0.841**	0.356	0.525	0.596	**0.808**
Contemporary	0.657	0.778	0.773	0.839	**0.454**	0.535	0.574	0.778	**0.418**	**0.636**	**0.630**	0.804

7. Discussion

Table 2 and 3 show that the systems fine-tuned on the German Newseye and Sonar corpora perform worse than those fine-tuned on the German HIPE2020 dataset. This could be because the used datasets differ significantly in size; Sonar and News- eye are much smaller than HIPE2020, so the lousy performance could come from overfitting. We also looked at each dataset's best- and worst-performing label and reported them with their F1-score in Tables 8 to 11.

The evaluation metrics (Tables 8 to 11) reveal that both the best model and the ensembled models better recognize the *LOC*, *PER*, and *TIME* labels across all datasets and measurements. While these labels dominate the corresponding datasets, hard-label ensembling ('voting') reflects the preference to reassign the label *O*s to these tokens. In contrast, the *ORG* and other minor category labels are generally worse handled, as a system is less likely to predict them, and in many cases, voting overrode these labels in favor of a more frequent predicted label. For example, the gold standard for a label is *ORG*. One model predicts the correct, infrequent label, the other 3 predict the more likely *O*. Due to majority voting, the correct guess is overruled and the incorrect label is chosen.

7.1. Comparison of Contemporary and Historical BERT

Table 6 demonstrates the Micro-F1 results for HIPE2020 German between the BERT-based model trained on historical data and the BERT-based model trained on contemporary data. The HIPE2020 data are from historical newspapers between the 19th and 20th centuries, as do the training data of the BERT-based HLM. As a result, the BERT-based model trained on historical data outperforms the BERT-based model trained on contemporary data on the overall result (column ALL in Table 6). The outcome is as expected because the pre-trained data cover historical data requirements in the historical NER task.

7.2. Error Analysis

To clarify the errors in our models, we conduct error analysis on the models trained on the Newseye dataset. We compare the NER results of the hard-label ensembling models on the Newseye French test-set (as the best-performing model) and the Newseye German test-set (as the worst-performing model) to the gold standard data. The five major errors discovered are as follows:

- **Right classification, wrong segmentation:** e.g. *B-PER* v.s. *I-PER*.
- **Wrong classification, right segmentation:** e.g. *B-LOC* v.s. *B-PER*.
- **Wrong classification, wrong segmentation:** The models predicted different NEs compared to the annotated data, e.g., *B-LOC* v.s. *I-PER*.
- **Complete false positive:** All tokens of a predicted entity are labeled with *O* in the gold-standard.
- **Complete false negative:** All Tokens of an entity in the gold-standard are predicted as *O*.

Figure 1 and 2 visualize the results of the error analysis. Besides the **Complete false positive** and **Complete false negative** errors, the two most frequent errors are **Right classification, wrong segmentation** and **Wrong classification, wrong segmentation**.

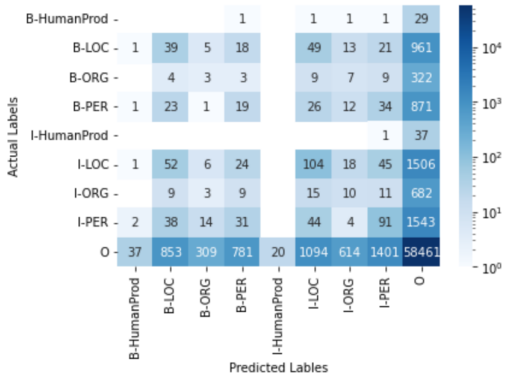

Figure 1: NER performance of the hard-label ensembling model on the Newseye French test-set.

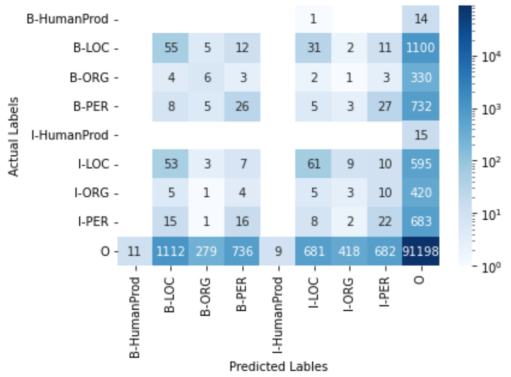

Figure 2: NER performance of the hard-label ensembling model using the Newseye German test-set.

The most common errors appear to follow a pattern. They usually occur at the incorrect entity recognition of the beginning token. To wrong segmentation errors, our model occasionally labels just the entities of the names without their titles because it fails to recognize the beginning

tokens of location or individual titles such as 'café' or 'v' '.'. Subword tokenization also raises segmentation errors particularly in the French NER task. The model typically performs NER and labels the tokens without the negation symbol '¬'.

With an incorrectly labeled entity classification to the beginning token, the remaining sequence of tokens follows the incorrect classification. We see this error randomly happen in sentences. Our model tends to start the NE with a non-location or non-personal noun, a punctuation, or an article. After assigning the beginning token, the following tokens will adopt the same incorrect classification. The following examples explain the two most common errors.

Right classification, wrong segmentation The French gold standard data label 'café' 'Mollard' as '*B-LOC*' '*I-LOC*', whereas our model labels 'café' 'Mollard' as '*O*' '*I-LOC*'. Without labeling 'café', our model assigns 'Mollard' as a beginning token.

In the German gold standard data, the entities of the German family name 'v' '.' 'Plener' are '*B-PER*' '*I-PER*' '*I-PER*'. They exist several times in the dataset, but not all have been correctly recognized. Although our model adequately recognized the first two occurrences of the three tokens, it does not consistently learn the entity pattern. Thus, there are also errors such as just 'Plener' as '*B-PER*' or 'v' as '*B-PER*'.

Subword tokenization, for instance, 'Rau¬' 'court', is challenging for our model to perform exactly NER. Our model predicts the entities as '*O*' '*B-LOC*' instead of '*B-LOC*' '*I-LOC*' as in the French gold standard data. Other examples include 'Ro¬' 'mans' (gold NER as '*I-PER*' '*I-PER*'), 'Pierre¬' 'Vaast' (gold NER as '*I-LOC*' '*I-LOC*') or 'AI¬' 'bert' (gold NER as '*I-PER*' '*I-PER*') which our model does not recognize all tokens containing '¬' and assigns '*O*' to them.

Wrong classification, wrong segmentation With an inanimate French noun like 'matinée' ('morning' in French), our model recognizes its entity classification as '*B-PER*', which should not be labelled. The incorrect classification leads the following tokens '.' '—' 'Le' 'président' 'du' 'Conseil' 'a' 'reçu' ('. - the president of the council has received' in French) all in '*I-PER*' where only 'Conseil' should be recognized as '*B-ORG*'.

Our model recognizes all possible tokens including punctuation to perform NER. For instance, only 'professeur' 'Vaquez' have entities of '*B-PER*' '*I-PER*' from the token sequence: '","' 'nièce' 'du' 'professeur' 'Vaquez' '.' (', niece of professor Vaquez.' in French). However, our NER model begins with the entity '*B-LOC*' by '","' and the following tokens are '*I-LOC*'.

Based on these identified errors, we are encouraged to propose relabeling post-submission experiments to improve the NERC task.

7.3. Post-Submission Experiments

Soft-Labeling With improvements across all scores, soft-label ensembling is preferable to hard-label ensembling. It seems that the additional information embedded in soft-labeling benefits the system. However, to be able to make more profound statements extending to other datasets, more experiments would be needed.

Multilingual Models Multilingual approaches did not improve the performance of our system. However, their performance is comparable to our monolingual approaches. The best

multilingually trained model performed better than the average best single monolingual model.

However, it seems that the performance of multilingual ensemble predictions could still be improved through a better selection of multilingual models or leveraging newer models such as XLM-R [20]. In addition, it is striking that all models performed poorly in German; more analyses should be done to investigate further reasons for this and improve the system.

Relabeling Relabeling improves Precision scores and deteriorates Recall scores. This means that our systems tend to return very few but precise NE predictions. Relabeling would be suited best for scenarios where precision is more important than recall, and False positives should be avoided.

8. Future Work

The existing system's performance could be optimized by using early stopping instead of training for a fixed number of epochs and applying grid-search on hyper-parameters.

We investigated the influence of frozen and unfrozen embeddings anecdotally, revealing that employing frozen word embeddings for NER tasks slightly improved results. However, due to time constraints, we could not implement our system with frozen embeddings, which would assumedly improve the overall results, especially for the relatively small training sets we used.

Because of the workflow in our experiment, each dataset and languages use different pre-trained models. Using the same models for one language across datasets and using multilingual models across languages would make the system more uniform and easier to improve as a whole.

Our ensembling approach could benefit from using a more careful selection of the single models, and replacing the worst model would probably improve the overall performance. This could particularly help the approach described in the post-submission experiment on multilingual models, where all models performed poorly in German. More analyses should be done to explain this bad performance and to improve it.

Experiments with other frameworks, such as AdapterHub [21] or other different architectures could improve performance. Other newer models, such as RoBERTa [15], XLM-R [20], or models trained on historical newspapers, could also help to improve performance.

9. Conclusion

In this paper, we reported on the performance of different language models for Named Entity Recognition (NER) in historical newspapers. One of the main challenges in this domain is digitization artifacts, a problem we address by fine-tuning models which have already been pretrained on noisy historical data. Furthermore, we experiment with ensembling, multilingual models, and label simplification.

In a case study for all languages of the HIPE-CLEF 2022 Newseye dataset, we found that models that have been trained over all languages did not improve the scores compared to monolingual models. In a second case study for the Newseye French dataset, we found that solely predicting entity categories and inferring the IOB encoding in postprocessing did not

help to improve F1- measures but shifted the scores to higher precision and a lower recall. On the same dataset, soft-label ensembling substantially improved all scores compared to hard-label ensembling.

Acknowledgments

Thanks to Simon Clematide and Andrianos Michail for lecturing the courses "Machine Learning for NLP 1 and 2" and for mentoring our group during development.

References

[1] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, A. M. Rush, HuggingFace's Transformers: State-of-the-art Natural Language Processing, Technical Report arXiv:1910.03771, arXiv, 2020. URL: http://arxiv.org/abs/1910.03771. doi:10.48550/arXiv.1910.03771, arXiv:1910.03771 [cs] type: article.

[2] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, 2017. arXiv:1706.03762.

[3] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. Le Scao, S. Gugger, M. Drame, Q. Lhoest, A. Rush, Transformers: State-of-the-art natural language processing, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics, Online, 2020, pp. 38–45. URL: https://aclanthology.org/2020.emnlp-demos.6. doi:10.18653/v1/2020.emnlp-demos.6.

[4] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, 2018. URL: https://arxiv.org/abs/1810.04805. doi:10.48550/ARXIV.1810.04805.

[5] M. Ehrmann, A. Hamdi, E. Linhares Pontes, M. Romanello, A. Doucet, Named entity recognition and classification on historical documents: A survey, arXiv e-prints (2021) arXiv–2109.

[6] J. Li, A. Sun, J. Han, C. Li, A survey on deep learning for named entity recognition, IEEE Transactions on Knowledge and Data Engineering 34 (2022) 50–70. doi:10.1109/TKDE.2020.2981314.

[7] S. Ghannay, C. Grouin, T. Lavergne, Experiments from limsi at the french named entity recognition coarse-grained task, in: Proc of CLEF 2020 LNCS, Thessaloniki, Greece, 2020.

[8] L. Martin, B. Muller, P. J. Ortiz Suárez, Y. Dupont, L. Romary, É. de la Clergerie, D. Seddah, B. Sagot, CamemBERT: a tasty French language model, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 7203–7219. URL: https://aclanthology.org/2020.acl-main.645. doi:10.18653/v1/2020.acl-main.645.

[9] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized BERT pretraining approach, CoRR abs/1907.11692 (2019). URL: http://arxiv.org/abs/1907.11692. arXiv:1907.11692.

[10] K. Todorov, G. Colavizza, Transfer learning for named entity recognition in historical corpora, in: CLEF (Working Notes), 2020. URL: http://ceur-ws.org/Vol-2696/paper_168.pdf.

[11] V. Provatorova, S. Vakulenko, E. Kanoulas, K. Dercksen, J. M. van Hulst, Named entity recognition and linking on historical newspapers: Uva.ilps & rel at clef hipe 2020, in: L. Cappellato, C. Eickhoff, N. F. 0001, A. Névéol (Eds.), Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum, Thessaloniki, Greece, September 22-25, 2020, volume 2696 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2020. URL: http://ceur-ws.org/Vol-2696/paper_209.pdf.

[12] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, Technical Report arXiv:1810.04805, arXiv, 2019. URL: http://arxiv.org/abs/1810.04805. doi:10.48550/arXiv.1810.04805, arXiv:1810.04805 [cs] type: article.

[13] M. Ehrmann, M. Romanello, A. Doucet, S. Clematide, Hipe 2022 shared task participation guidelines v1.0, 2022. URL: https://doi.org/10.5281/zenodo.6045662.

[14] B. Staatsbibliothek, dbmdz (Bayerische Staatsbibliothek), 2022. URL: https://huggingface.co/dbmdz.

[15] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, RoBERTa: A Robustly Optimized BERT Pretraining Approach, Technical Report arXiv:1907.11692, arXiv, 2019. URL: http://arxiv.org/abs/1907.11692. doi:10.48550/arXiv.1907.11692, arXiv:1907.11692 [cs] type: article.

[16] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, Q. V. Le, XLNet: Generalized Autoregressive Pretraining for Language Understanding, Technical Report arXiv:1906.08237, arXiv, 2020. URL: http://arxiv.org/abs/1906.08237. doi:10.48550/arXiv.1906.08237, arXiv:1906.08237 [cs] type: article.

[17] K. Clark, M.-T. Luong, Q. V. Le, C. D. Manning, ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators, Technical Report arXiv:2003.10555, arXiv, 2020. URL: http://arxiv.org/abs/2003.10555. doi:10.48550/arXiv.2003.10555, arXiv:2003.10555 [cs] type: article.

[18] L. Tunstall, L. von Werra, T. Wolf, Natural Language Processing with Transformers, O'Reilly, 2022.

[19] C. Ju, A. Bibaut, M. Laan, The relative performance of ensemble methods with deep convolutional neural networks for image classification, Journal of applied statistics (2018). doi:10.1080/02664763.2018.1441383.

[20] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, É. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised cross-lingual representation learning at scale, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020, pp. 8440–8451.

[21] J. Pfeiffer, A. Rücklé, C. Poth, A. Kamath, I. Vulic, S. Ruder, K. Cho, I. Gurevych, Adapterhub: A framework for adapting transformers, CoRR abs/2007.07779 (2020). URL: https://arxiv.org/abs/2007.07779. arXiv:2007.07779.

10. Online Resources

- Repository for this paper,
- HIPE2022 datasets,
- HIPE2022 evaluation module,
- Google Colab.

Table 7
Used models and their corresponding HuggingFace links. Not available dev-set scores are marked with '-'. The best model for each dataset is marked in **bold**. The best multilingual model was determined by the test-set.

Languages	Datasets	Model name (hyperlink)	F1-macro on dev-set
De	Newseye	bert-base-german-cased	0.35
De	Sonar		0.87
Fi	Newseye	dbmdz/bert-base-finnish-europeana-cased	0.78
Fr	HIPE2020	dbmdz/bert-base-french-europeana-cased	0.82
Fr	**Newseye**		**0.86**
Fr	Letemps		0.55
De	HIPE2020	dbmdz/bert-base-german-europeana-cased	0.76
De	**Newseye**		**0.46**
De	**Sonar**		**0.93**
En	Topres19th	dbmdz/bert-base-historic-english-cased	0.62
En	HIPE2020		-
De	HIPE2020	dbmdz/bert-base-historic-multilingual-cased	0.74
De	Sonar		0.60
En	Topres19th		0.72
Fi	Newseye		0.80
Fr	HIPE2020		0.80
Fr	Newseye		0.80
Fr	**Letemps**		**0.56**
Sv	Newseye		0.71
Multilingual	**Newseye**		-
Multilingual	Newseye	bert-base-multilingual-cased	-
Multilingual	Newseye	bert-base-multilingual-uncased	-
Sv	Newseye	dbmdz/bert-base-swedish-europeana-cased	0.73
Multilingual	Newseye	distilbert-base-multilingual-cased	-
Fr	HIPE2020	dbmdz/electra-base-french-europeana-cased-discriminator	0.34
Fr	Newseye		0.38
Fr	Letemps		0.32
De	**HIPE2020**	dbmdz/electra-base-german-europeana-cased-discriminator	**0.77**
Fr	HIPE2020	dbmdz/flair-hipe-2022-ajmc-fr-64k	-
Fi	**Newseye**	EMBEDDIA/finest-bert	**0.83**
En	**Topres19th**	google/electra-base-discriminator	**0.73**
En	HIPE2020		-
En	Topres19th	Jean-Baptiste/roberta-large-ner-english	0.71
En	HIPE2020		-
Sv	Newseye	jonfd/electra-small-nordic	0.13
Sv	**Newseye**	KB/bert-base-swedish-cased	**0.73**
Fi	Newseye	setu4993/LaBSE	0.79
Fr	Newseye		0.84
Fr	Letemps		0.52
Fr	**HIPE2020**		**0.83**
Sv	Newseye		0.66
Multilingual	Newseye		-
Fi	Newseye	TurkuNLP/bert-base-finnish-cased-v1	0.79
En	Topres19th	xlnet-base-cased	0.26
En	HIPE2020		-

Table 8
Macro-fuzzy evaluation of submitted systems

System	Dataset	Eval	Precision all	Recall all	F1 all-labels	Best category	Worst category
Best Model	HIPE2020_de	macro-fuzzy	0.789	0.828	0.796	TIME (0.9)	ORG (0.484)
Best Model	HIPE2020_fr	macro-fuzzy	0.865	0.831	0.836	TIME (0.942)	ORG (0.587)
Best Model	Letemps_fr	macro-fuzzy	0.52	0.712	0.752	PERS (0.845)	ORG (0.256)
Best Model	Newseye_de	macro-fuzzy	0.392	0.502	0.547	PER (0.591)	HUMANPROD (0.0)
Best Model	Newseye_fi	macro-fuzzy	0.765	0.626	0.668	HUMANPROD (0.863)	ORG (0.57)
Best Model	Newseye_fr	macro-fuzzy	0.775	0.777	0.766	PER (0.82)	ORG (0.606)
Best Model	Newseye_sv	macro-fuzzy	0.738	0.72	0.735	LOC (0.821)	ORG (0.428)
Best Model	Sonar_de	macro-fuzzy	0.617	0.667	0.633	LOC (0.711)	ORG (0.451)
Best Model	Topres19th_en	macro-fuzzy	0.813	0.86	0.824	LOC (0.873)	BUILDING (0.643)
AVERAGE			**0.62**	**0.63**	**0.64**		
Ensembled	HIPE2020_de	macro-fuzzy	0.819	0.826	0.808	TIME (0.89)	ORG (0.501)
Ensembled	HIPE2020_en	macro-fuzzy	0.724	0.656	0.689	TIME (1.0)	PROD (0.0)
Ensembled	HIPE2020_fr	macro-fuzzy	0.858	0.816	0.826	TIME (0.951)	ORG (0.609)
Ensembled	Letemps_fr	macro-fuzzy	0.545	0.712	0.763	LOC (0.847)	ORG (0.239)
Ensembled	Newseye_de	macro-fuzzy	0.403	0.469	0.538	LOC (0.585)	HUMANPROD (0.167)
Ensembled	Newseye_fi	macro-fuzzy	0.796	0.581	0.703	HUMANPROD (0.795)	ORG (0.593)
Ensembled	Newseye_fr	macro-fuzzy	0.814	0.762	0.779	PER (0.811)	ORG (0.621)
Ensembled	Newseye_sv	macro-fuzzy	0.765	0.722	0.747	HUMANPROD (0.861)	ORG (0.417)
Ensembled	Sonar_de	macro-fuzzy	0.663	0.672	0.654	LOC (0.758)	ORG (0.432)
Ensembled	Topres19th_en	macro-fuzzy	0.881	0.823	0.841	LOC (0.889)	BUILDING (0.642)
AVERAGE			**0.70**	**0.68**	**0.69**		

Table 9
Macro-strict evaluation of submitted systems

System	Dataset	Eval	Precision all	Recall all	F1 all-labels	Best category	Worst category
Best Model	HIPE2020_de	macro-strict	0.671	0.693	0.672	LOC (0.805)	ORG (0.384)
Best Model	HIPE2020_fr	macro-strict	0.764	0.735	0.74	LOC (0.772)	ORG (0.499)
Best Model	Letemps_fr	macro-strict	0.448	0.625	0.659	LOC (0.754)	ORG (0.114)
Best Model	Newseye_de	macro-strict	0.302	0.386	0.421	LOC (0.464)	HUMANPROD (0.0)
Best Model	Newseye_fi	macro-strict	0.682	0.561	0.596	PER (0.733)	ORG (0.481)
Best Model	Newseye_fr	macro-strict	0.63	0.623	0.621	HUMANPROD (0.699)	ORG (0.419)
Best Model	Newseye_sv	macro-strict	0.611	0.583	0.602	HUMANPROD (0.758)	ORG (0.338)
Best Model	Sonar_de	macro-strict	0.46	0.5	0.474	LOC (0.649)	ORG (0.241)
Best Model	Topres19th_en	macro-strict	0.77	0.812	0.779	LOC (0.824)	BUILDING (0.527)
AVERAGE			**0.62**	**0.63**	**0.64**		
Ensembled	HIPE2020_de	macro-strict	0.686	0.679	0.67	LOC (0.815)	ORG (0.411)
Ensembled	HIPE2020_en	macro-strict	0.553	0.494	0.523	TIME (0.718)	PROD (0.0)
Ensembled	HIPE2020_fr	macro-strict	0.741	0.7	0.712	LOC (0.712)	ORG (0.403)
Ensembled	Letemps_fr	macro-strict	0.48	0.636	0.681	LOC (0.776)	ORG (0.095)
Ensembled	Newseye_de	macro-strict	0.316	0.364	0.419	LOC (0.494)	HUMANPROD (0.167)
Ensembled	Newseye_fi	macro-strict	0.666	0.484	0.585	LOC (0.655)	ORG (0.477)
Ensembled	Newseye_fr	macro-strict	0.659	0.614	0.63	HUMANPROD (0.766)	ORG (0.471)
Ensembled	Newseye_sv	macro-strict	0.654	0.608	0.634	HUMANPROD (0.739)	ORG (0.306)
Ensembled	Sonar_de	macro-strict	0.512	0.514	0.503	LOC (0.695)	ORG (0.23)
Ensembled	Topres19th_en	macro-strict	0.839	0.785	0.802	LOC (0.86)	BUILDING (0.554)
AVERAGE			**0.69**	**0.68**	**0.69**		

Table 10
Micro-fuzzy evaluation of submitted systems

System	Dataset	Eval	Precision all	Recall all	F1 all-labels	Best category	Worst category
Best Model	HIPE2020_de	micro-fuzzy	0.783	0.826	0.804	PERS (0.874)	ORG (0.545)
Best Model	hipe2020_fr	micro-fuzzy	0.825	0.776	0.8	PERS (0.848)	PROD (0.596)
Best Model	Letemps_fr	micro-fuzzy	0.61	0.771	0.681	LOC (0.734)	ORG (0.208)
Best Model	Newseye_de	micro-fuzzy	0.48	0.512	0.495	LOC (0.541)	HUMANPROD (0.0)
Best Model	Newseye_fi	micro-fuzzy	0.681	0.603	0.64	HUMANPROD (0.732)	ORG (0.478)
Best Model	Newseye_fr	micro-fuzzy	0.785	0.787	0.786	PER (0.849)	HUMANPROD (0.579)
Best Model	Newseye_sv	micro-fuzzy	0.786	0.704	0.742	LOC (0.799)	ORG (0.457)
Best Model	Sonar_de	micro-fuzzy	0.625	0.718	0.668	LOC (0.765)	ORG (0.468)
Best Model	Topres19th_en	micro-fuzzy	0.807	0.851	0.829	LOC (0.872)	STREET (0.661)
AVERAGE			**0.61**	**0.63**	**0.65**		
Ensembled	HIPE2020_de	micro-fuzzy	0.812	0.833	0.822	LOC (0.866)	PROD (0.574)
Ensembled	HIPE2020_en	micro-fuzzy	0.726	0.661	0.692	TIME (0.909)	PROD (0.0)
Ensembled	HIPE2020_fr	micro-fuzzy	0.824	0.773	0.798	TIME (0.847)	ORG (0.555)
Ensembled	Letemps_fr	micro-fuzzy	0.642	0.773	0.701	LOC (0.7)	ORG (0.178)
Ensembled	Newseye_de	micro-fuzzy	0.481	0.478	0.479	LOC (0.551)	HUMANPROD (0.08)
Ensembled	Newseye_fi	micro-fuzzy	0.73	0.619	0.67	PER (0.706)	ORG (0.495)
Ensembled	Newseye_fr	micro-fuzzy	0.801	0.744	0.772	PER (0.839)	ORG (0.58)
Ensembled	Newseye_sv	micro-fuzzy	0.797	0.702	0.746	LOC (0.801)	ORG (0.442)
Ensembled	Sonar_de	micro-fuzzy	0.641	0.696	0.667	LOC (0.78)	ORG (0.443)
Ensembled	Topres19th_en	micro-fuzzy	0.869	0.81	0.838	LOC (0.88)	BUILDING (0.659)
AVERAGE			**0.70**	**0.67**	**0.68**		

Table 11
Micro-strict evaluation of submitted systems

System	Dataset	Eval	Precision all	Recall all	F1 all-labels	Best category	Worst category
Best Model	HIPE2020_de	micro-strict	0.677	0.714	0.695	LOC (0.794)	ORG (0.411)
Best Model	HIPE2020_fr	micro-strict	0.718	0.675	0.696	LOC (0.748)	PROD (0.519)
Best Model	Letemps_fr	micro-strict	0.557	0.704	0.622	LOC (0.692)	ORG (0.12)
Best Model	Newseye_de	micro-strict	0.395	0.421	0.408	LOC (0.479)	HUMANPROD (0.0)
Best Model	newseye_fi	micro-strict	0.592	0.524	0.556	HUMANPROD (0.683)	ORG (0.407)
Best Model	Newseye_fr	micro-strict	0.655	0.657	0.656	PER (0.709)	ORG (0.441)
Best Model	Newseye_sv	micro-strict	0.673	0.603	0.636	HUMANPROD (0.75)	ORG (0.343)
Best Model	Sonar_de	micro-strict	0.447	0.513	0.477	LOC (0.685)	ORG (0.293)
Best Model	Topres19th_en	micro-strict	0.761	0.802	0.781	LOC (0.833)	BUILDING (0.564)
AVERAGE			**0.64**	**0.66**	**0.67**		
Ensembled	HIPE2020_de	micro-strict	0.716	0.735	0.725	LOC (0.82)	PROD (0.452)
Ensembled	HIPE2020_en	micro-strict	0.538	0.49	0.513	LOC (0.607)	PROD (0.0)
Ensembled	HIPE2020_fr	micro-strict	0.7	0.657	0.678	LOC (0.761)	PROD (0.421)
Ensembled	Letemps_fr	micro-strict	0.589	0.71	0.644	LOC (0.715)	ORG (0.089)
Ensembled	Newseye_de	micro-strict	0.396	0.394	0.395	LOC (0.485)	HUMANPROD (0.08)
Ensembled	Newseye_fi	micro-strict	0.618	0.524	0.567	HUMANPROD (0.615)	ORG (0.385)
Ensembled	Newseye_fr	micro-strict	0.673	0.625	0.648	PER (0.712)	ORG (0.455)
Ensembled	Newseye_sv	micro-strict	0.686	0.604	0.643	LOC (0.716)	ORG (0.288)
Ensembled	Sonar_de	micro-strict	0.47	0.511	0.49	LOC (0.709)	ORG (0.268)
Ensembled	Topres19th_en	micro-strict	0.816	0.76	0.787	LOC (0.84)	BUILDING (0.551)
AVERAGE			**0.69**	**0.66**	**0.67**		

Table 12
Evaluation multilingual experiments. Model1: dbmdz/bert-base-historic-multilingual-cased, Model2: setu4993/LaBSE, Model3: bert-base-multilingual-cased, Model4: bert-base-multilingual-uncased, Model5: distilbert-base-multilingual-cased

System	Dataset	Eval	Precision all	Recall all	F1 all-labels
sub_Best Model	Newseye_de	micro-strict	0.395	0.421	0.408
sub_Best Model	Newseye_fi	micro-strict	0.592	0.524	0.556
sub_Best Model	Newseye_fr	micro-strict	0.655	0.657	0.656
sub_Best Model	Newseye_sv	micro-strict	0.673	0.603	0.636
AVERAGE			**0.54**	**0.52**	**0.56**
sub_Ensembled	Newseye_de	micro-strict	0.396	0.394	0.395
sub_Ensembled	Newseye_fi	micro-strict	0.618	0.524	0.567
sub_Ensembled	Newseye_fr	micro-strict	0.673	0.625	0.648
sub_Ensembled	Newseye_sv	micro-strict	0.686	0.604	0.643
AVERAGE			**0.70**	**0.65**	**0.67**
set_random	Newseye_de	micro-strict	0.006	0.022	0.009
set_random	Newseye_fi	micro-strict	0.005	0.01	0.007
set_random	Newseye_fr	micro-strict	0.005	0.013	0.008
set_random	Newseye_sv	micro-strict	0.01	0.023	0.013
AVERAGE			**0.01**	**0.02**	**0.01**
Model1	Newseye_de	micro-strict	0.407	0.399	0.403
Model1	Newseye_fi	micro-strict	0.671	0.564	0.613
Model1	Newseye_fr	micro-strict	0.653	0.616	0.634
Model1	Newseye_sv	micro-strict	0.729	0.627	0.674
AVERAGE			**0.62**	**0.55**	**0.58**
Model2	Newseye_de	micro-strict	0.406	0.416	0.411
Model2	Newseye_fi	micro-strict	0.624	0.514	0.563
Model2	Newseye_fr	micro-strict	0.65	0.607	0.628
Model2	Newseye_sv	micro-strict	0.693	0.599	0.643
AVERAGE			**0.59**	**0.53**	**0.56**
Model3	Newseye_de	micro-strict	0.407	0.44	0.423
Model3	Newseye_fi	micro-strict	0.586	0.462	0.517
Model3	Newseye_fr	micro-strict	0.648	0.585	0.615
Model3	Newseye_sv	micro-strict	0.643	0.548	0.592
AVERAGE			**0.57**	**0.51**	**0.54**
Model4	Newseye_de	micro-strict	0.405	0.428	0.416
Model4	Newseye_fi	micro-strict	0.563	0.438	0.493
Model4	Newseye_fr	micro-strict	0.626	0.587	0.606
Model4	Newseye_sv	micro-strict	0.637	0.53	0.579
AVERAGE			**0.56**	**0.50**	**0.52**
Model5	Newseye_de	micro-strict	0.232	0.157	0.188
Model5	Newseye_fi	micro-strict	0.266	0.113	0.159
Model5	Newseye_fr	micro-strict	0.245	0.239	0.242
Model5	Newseye_sv	micro-strict	0.356	0.182	0.241
AVERAGE			**0.27**	**0.17**	**0.21**
Ensembling	Newseye_de	micro-strict	0.434	0.408	0.421
Ensembling	Newseye_fi	micro-strict	0.649	0.502	0.566
Ensembling	Newseye_fr	micro-strict	0.691	0.608	0.647
Ensembling	Newseye_sv	micro-strict	0.742	0.596	0.661
AVERAGE			**0.629**	**0.5285**	**0.57375**

Table 13
Relabeling post-submission experiment results

Model	Pretrained model	Evaluation setting	No relabeling			Relabeling		
			Precision	Recall	F1	Precision	Recall	F1
1	Language-agnostic BERT Sentence Encoder (LaBSE)	micro - fuzzy	0.772	0.724	0.747	**0.782**	0.699	0.738
2	Historic Language Multilingual Models		0.77	0.734	0.752	**0.778**	0.72	0.748
3	French Europeana BERT		0.785	0.787	0.786	**0.806**	0.765	0.785
4	French Europeana ELECTRA		0.468	0.521	0.493	**0.47**	**0.544**	**0.504**
1	Language-agnostic BERT Sentence Encode (LaBSE)	micro - strict	0.645	0.605	0.624	**0.654**	0.585	0.618
2	Historic Language Multilingual Models		0.651	0.621	0.636	**0.661**	0.611	0.635
3	French Europeana BERT		0.655	0.657	0.656	**0.672**	0.638	0.654
4	French Europeana ELECTRA		0.213	0.238	0.225	**0.233**	**0.269**	**0.25**