LIT: Label-Informed Transformers on Token-Based Classification

Wenjun Sun, Hanh Thi Hong Tran, Carlos-Emiliano González-Gallardo, Mickaël Coustaty, Antoine Doucet

LIT: Label-Informed Transformers on Token-based Classification

Wenjun Sun[1][0009−0002−7857−8737], Hanh Thi Hong Tran[1,2,3][0000−0002−5993−1630], Carlos-Emiliano González-Gallardo[1,4][0000−0002−0787−2990], Mickaël Coustaty[1][0000−0002−0123−439X], and Antoine Doucet[1][0000−0001−6160−3356]

[1] L3i, University of La Rochelle, La Rochelle, France
{firstname.lastname,thi.tran}@univ-lr.fr
[2] Jožef Stefan International Postgraduate School, Ljubljana, Slovenia
[3] Jožef Stefan Institute, Ljubljana, Slovenia
[4] LIFAT, University of Tours, Blois, France
carlos-emiliano.gonzalez-gallardo@univ-tours.fr

Abstract. Transformer-based language models have led to the investigation of various embedding and modeling techniques for several downstream natural language processing tasks. Nevertheless, the comprehensive exploration of semantic information about the label from encoder and decoder components in these tasks is yet to be fully realized. In this paper, we propose LIT, an end-to-end pipeline architecture that integrates the transformer's encoder-decoder mechanism with an additional label semantic to token classification tasks (i.e., historical named entity recognition (NER) and automatic term extraction (ATE)). Our findings demonstrate that LIT outperforms the benchmark in F1 with a maximal rise of 9.5 percentage points in the historical NER task and 11.2 percentage points in the ATE task for the gold standard excluding named entities.

Keywords: LIT · Transformers · Label Semantic Similarity · BERT · LLaMA · NER · Term Extraction

1 Introduction

The progress made in transformer-based language models has prompted exploration into diverse embedding and modeling techniques for multiple downstream tasks in natural language processing (NLP). However, there is still a need to fully explore the semantic information related to the label from the encoder and decoder components in these tasks. This highlights a promising avenue for research and development to leverage the potential benefits that such exploration may bring to the field of NLP.

In this paper, we propose LIT, a novel architecture that integrates the transformer's encoder-decoder mechanism with an additional label semantic similarity. While the encoder is responsible for providing the semantic embedding, the

decoder calculates the corresponding semantics of each label based on the preset labels. Finally, each token label is assigned by comparing each token output's embeddings from the encoder with the entity semantics output from the decoder. The number of labels for the decoder is determined based on the preset number of labels in the dataset, and the specific weights of these queries are randomly initialized and continuously updated during training.

We employ an end-to-end pipeline for multiple NLP downstream tasks, including two kinds of token-based classification tasks such as historical named entity recognition (NER) and automatic term extraction (ATE). We performed a thorough error analysis for each downstream task and found that: (1) Our model excels at identifying terms in the ATE task and accurately determining the corresponding types of named entities (NEs) in historical NER. However, it requires further improvement in recognizing non-NEs and non-terms, which leads to lower performance in terms of precision; (2) large language models (LLMs) are less capable than traditional language models on historical NER tasks. Although our end-to-end system does not outperform the task-specific model in all metrics, it is a good start to designing a general model. Overall, the system is well-suited to multiple tasks (most results have outperformed the benchmarks). Our code is available through the code repository[5].

The rest of this paper is structured as follows: in Section 2 we examine prior work with BERT-based language models. Our proposed end-to-end architecture is outlined in Section 3, followed by a detailed implementation along with a dataset description in Section 4. The findings of our experiments are presented and discussed in Section 5. Finally, conclusions and plans for future work are drawn in Section 6.

2 Related Work

Recent years have witnessed the evolution of fine-tuning the BERT model [15] or its variants [10,13,31] to enhance model performance with benchmark pretrained models, especially fine-tuning for specific tasks such as hmBERT [29] for historical NER where named entities are extracted and classified from historical texts, LeBERT [21] for sentiment analysis where emotions or attitudes are identified given a text sequence, or other BERT variants (e.g., XLMR [36,33,35], InfoXLM [34]) for the ATE task (see [32] for details) where domain-specific terms are detected from texts without manual intervention.

Additional information has also been adapted to enrich the input representation that is fed into the BERT-based model, such as lexical information [16], local and global information [20], knowledge graphs [41,43], and the position information [38], to mention a few. At the same time, there has been a lot of work attempting to add domain-specific knowledge to the BERT model to improve its processing power for specific tasks, e.g., knowledge of consumer sentiment has been added to the language model for text classification [14] and

[5] `https://github.com/WenjunSUN1997/ner_tr`

the additional supporting text has been exploited [42]. At the same time, several NER-specific approaches have arisen, such as analyzing multiple levels of information to improve the model's performance for NER [40], using structured inference to enhance semantics [39], and so on. When performing NER tasks on historical texts, the semantic changes caused by temporal changes are a big challenge for models trained and tested with modern corpora [9]. Moreover, the errors in text recognition of historical texts can have an impact on the language model extraction token embedding. Since the current mainstream methods are based on pre-trained models, the historical corpus has a great influence on the quality of semantic vectors of language models, but compared to the NER corpus of the modern corpus, there is a lack of corpus for historical data, which becomes a major limitation of historical text NER [4]. To address the above challenges, the researchers first used a rule-based approach to analyze the text by establishing definitions of the different named entities [3], then there are also machine learning-based methods, such as making the classification tasks with CRF [26] and introducing voting strategies into the classification [37]. To address the digitization errors of historical documents which can cause semantic bias, the researchers proposed a method to mitigate by stacking the encoder blocks of the transformer structure [2].

After autoencoder models (e.g., BERT and its variants), more autoregressive and generative LLMs such as GPT-3 [8] and T5 [23] have emerged. As computational resources and training data have expanded, the number of parameters for training these models has also increased. Meanwhile, LLMs such as LLaMA [30] and reinforcement learning with human feedback models like GPT-4 [22] and the Falcon Series of Language Models [1] have demonstrated capabilities that far exceed those of previous models. Still, due to their sheer size and computational resources, they are difficult to deploy. There is work that demonstrates the inability of LLMs to obtain the desired results in historical NER tasks [9].

3 Model

The general LIT architecture is shown in Figure 1. It consists of a backbone language model, an encoder, a feature extractor, a decoder, and a cosine similarity operation. The first three are used to compute the embedding of the target token, the decoder is used to obtain the embedding of the individual labels, and finally, the cosine similarity is used to obtain the final prediction. This architecture is then adapted for NER and ATE tasks.

For these two tasks, we propose the following end-to-end mechanism. First, when the training data is loaded, the input is represented by a sequence of tokens. Then, the vector representation of each token is obtained by the language model and fed directly to the encoder for further computation. We use this encoder for further processing of the semantics (e.g., historical data, domain-specific terms). During the tokenization process some tokens are divided into multiple sub-tokens, we thus adopt the vector representation of the first sub-token as the overall feature of this token. Then, the average vector value of each

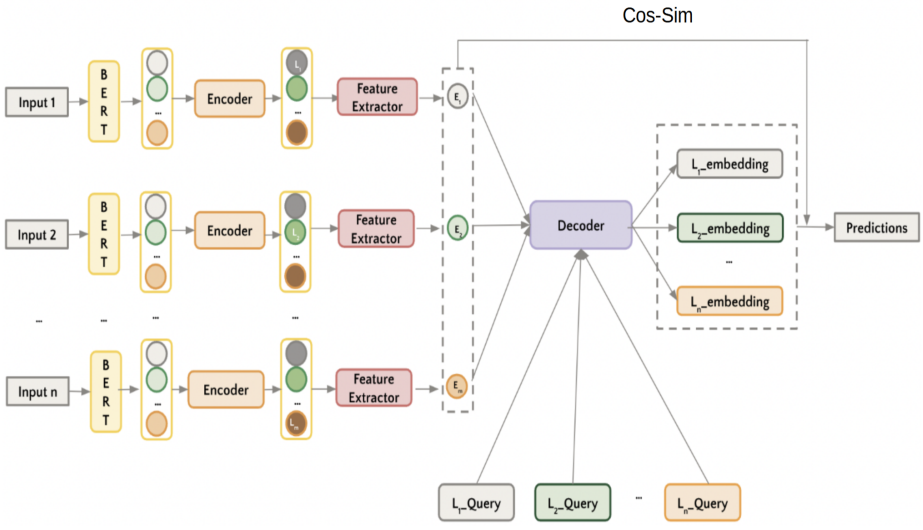

Fig. 1. General architecture of LIT

target token is extracted as the input of the decoder. Mathematically speaking, given the original sequence $\{T_1, T_2, ..., T_x\}$, the tokenization process returns the output sequence of $\{T_{1_1}, T_{1_2}, ..., T_{x_y}\}$, where T_{x_y} means the y^{th} sub-token of the x^{th} token. After BERT and the encoder, the resulting token embeddings are $\{Em_{1_1}, Em_{1_2}, ..., Em_{x_y}\}$, where Em_{x_y} refers to the embedding of the y^{th} sub-token of the x^{th} token and $L(Em)$ is the label of Em_x. For the named entity label PER (person), the representation of PER can be formulated as:

$$PER = Avg(\mathbf{1}_{\{L(Em_{i_1})==PER\}} Em_{i_1}); i = \{0, \ldots, x\} \tag{1}$$

Then, sequences are grouped into corresponding input groups according to the labels they contain. If one sequence contains multiple labels, it will be organized into multiple corresponding groups. This is to allow the model to learn the semantics of all labels during training. These representations are fed into the decoder as a memory along with n decoding vectors as another input to the decoder according to the number of labels. Finally, the n outputs of the decoder are obtained and used as the representation of each label. The predicted token label is the one that maximizes the cosine similarity (Cos_sim) between the encoder and decoder outputs, it is defined as:

$$L(Em_{x_{s1}}) = Max(Cos\_sim(Em_{x_{s1}}, Label))) \tag{2}$$

where $Label$ means the embeddings of labels. This mechanism is shown in Figure 2 for both token-based classification tasks. During inference, the feature extractor is not needed, as the final result is obtained by comparing the output of the encoder with the embedding of the label.

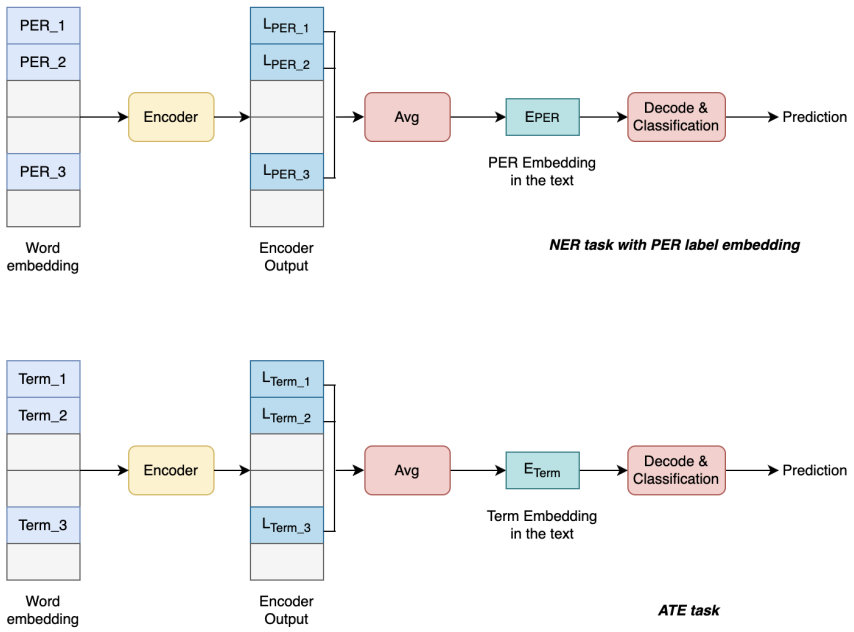

Fig. 2. Architecture adaptation on the token-based classification

4 Experiment

Regarding token-based classification, we focus on two tasks: historical NER and ATE. For historical NER, we experimented with the NewsEye dataset [11] in Finnish, Swedish, and French; along with the HIPE-2020 English dataset [5]. For ATE, we experimented with the ACTER dataset [25] in English and Dutch on four different domains.

4.1 Datasets

Historical NER

The NewsEye dataset [11] was collected through the national libraries of France, Austria, and Finland. The Finnish and Swedish corpora are composed of digitized newspapers published between 1848 and 1918. Both datasets are annotated with the BIO labeling scheme based on Quaero's guidelines [27] and contain four entity types: person (*PER*) for individuals or groups of people, location (*LOC*) for entities related to addresses or geographic locations, organization (*ORG*) for various types of groups or organizations, and human production (*HUMAN-PROD*) for media products.

The HIPE-2020 dataset [5] is a collection of Swiss, Luxembourgish, and American digitized newspapers in French, German, and English. It follows a

Table 1. Statistics of the historical NER datasets

Labels	NewsEye						HIPE-2020	
	Finnish		Swedish		French		English	
	Train	Test	Train	Test	Train	Test	Train	Test
PER	2,626	1,228	2,966	654	14,499	2,880	1,379	599
LOC	1,498	408	1,562	457	9,343	2,872	758	335
ORG	791	124	497	158	3,905	1,103	390	295
HUMANPROD	295	51	463	62	378	71	97	63
TIME	-	-	-	-	-	-	151	77

similar BIO labeling scheme as NewsEye but incorporates the additional label time (*TIME*) to represent dates. The English portion covers the period ranging from 1790 to 2010. Table 1 shows the entity type distribution for all of the datasets.

Term Extraction

The ACTER dataset [25] is a collection of 12 corpora covering four domains (*Corruption*, *Dressage*, *Wind energy*, and *Heart failure*) in English, French, and Dutch. This dataset comprises two types of gold standards, one encompassing both terms and named entities (NES) and the second one exclusively comprising terms (ANN). Regarding the data volume and term distribution, the *Heart failure* domain contains many more unique terms compared to the other three domains. The detailed description of ACTER can be found in the TermEval competition [24]. In this paper, we focus mainly on English and Dutch and apply the same configuration as in the TermEval 2020 share task and related works [12,18,33] where two domains are used for training, one for validation and one for testing. The *Heart failure* domain of each language is considered the test set.

4.2 Baseline

Named Entity Recognition. Historical NER aims to extract and classify named entities from historical texts, such as ancient manuscripts, newspapers, or even digitized books. We compare our proposed architecture with the benchmarks, the best-performing approaches, and posterior adaptations from the HIPE competitions held in 2020 and 2022 [6,7].

- **Neur-bsl** [29]: It corresponds to the baseline from HIPE-2022 organizers which used the multilingual language model $XLM-R_{BASE}$ as the backbone. They labeled only the first sub-token to facilitate the alignment of model outputs and ground-truth and fine-tuned for 10 epochs.
- **Aauzh** [28]: As a participant in the HIPE-2022 challenge, this team proposed a fine-tuned sequence-labeling transformer-based model with default hyperparameters setting and trained each dataset for 3 epochs. At inference

time, they used the summing pooling strategy to aggregate subtoken-level NER labels into token-level labels with soft-label ensembling techniques.
- **WLV** [17]: This team introduced a BERT-based architecture with an additional Bi-LSTM layer to capture meaningful representation. This information is fed to a conditional random field CRF layer to obtain the labels corresponding to each token in the final predictions.
- **Stack-NER** [2]: This method proposed a stack transformer-based mechanism that includes a fine-tuned BERT encoder and several transformer blocks. Stack-NER has proved to increase performance in historical documents, while not degrading the performance over modern data.
- **LLaMA2**$_{linear}$: As the large language model receives more and more attention, we also used the LLaMA2 model in our experiments for testing. This approach uses the LLaMA2 model as a backbone and a linear layer to predict the kinds of individual tokens. We selected the 7B version of LLaMA2 as the backbone language model.

Term Extraction. ATE seeks to identify domain-specific terms within text corpora, eliminating the necessity for manual intervention. Our approach was influenced by the TermEval 2020 shared task [24], where we regarded the outcomes achieved by the winning teams and the subsequent enhancements available post-competition as our baselines.

- **TALN-LS2N** [12]: This winning team for the English set used BERT as a binary classifier, where a combination of n-grams and a sentence were used as an instance and the classifier needed to determine for each n-gram whether or not it was a term.
- **NLPLab_UQAM** [24]: This winning approach for Dutch used pretrained GloVe word embeddings fed into a Bi-LSTM-based neural architecture.
- **BERT**$_{Tran22}$ [34]: This approach considered the ATE task as a sequence-labeling problem and used BERT as a token classifier.
- **roBERT-base**$_{Tran22}$ [34]: Sharing the same mechanism with the BERT$_{Tran22}$ token classifier, this setting was specifically fine-tuned for the Dutch language.

While TALN-LS2N and NLPLab_UQAM were the winning solution during the TermEval 2020 challenge, we use BERT$_{Tran22}$ and roBERT-base$_{Tran22}$ as the benchmark to compare with our methods to see the impact of additional label semantic similarity.

4.3 Settings

To make long texts compatible with the BERT model, we set a window length and truncation stride to 100, and used the cross-entropy function for loss calculation. For the language model, we fine-tuned BERT based on the dataset and

specific task [6]. The encoder and decoder of our model are both in the transformer structure. We observed that regarding the historical NER task, non-NEs (O) weight had a notable impact on the model's precision. Thus, we ran a greedy search of weights between 0 and 20 and based on the experimental results, we assigned a weight of 15 to non-NEs in the cross-entropy loss function.

5 Results and Discussion

We first present and analyse the results of the two experiments, and then discuss our model in light of each task.

5.1 Historical NER

The results of historical NER over NewsEye and HIPE-2020 datasets in a strict (exact boundary matching) and a fuzzy boundary matching setting are shown in Table 2. For NewsEye we experimented with Finnish, Swedish, and French; while for HIPE-2020, we focused on the English portion[7].

Table 2. Comparative results of historical NER

Models	NewsEye									HIPE-2020		
	Finnish			Swedish			French			English		
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
fuzzy												
Aauzh [28]	**0.730**	0.619	0.670	**0.797**	0.702	0.746	0.785	0.787	0.786	**0.726**	0.661	0.692
WLV [17]	-	-	-	-	-	-	-	-	-	0.582	0.626	0.603
Neur-bsl [29]	0.715	0.812	0.760	0.675	0.836	0.747	0.755	0.805	0.779	0.564	0.695	0.623
Stack-NER [2]	0.675	0.737	0.704	0.716	0.808	0.759	**0.887**	0.821	**0.853**	0.577	0.699	0.632
LLaMA2$_{linear}$	0.516	0.546	0.530	0.339	0.651	0.446	0.288	0.689	0.406	0.354	0.576	0.439
LIT	0.678	**0.877**	**0.765**	0.708	**0.904**	**0.794**	0.567	**0.892**	0.694	0.608	**0.826**	**0.701**
strict												
Aauzh [28]	**0.618**	0.524	0.567	**0.686**	0.604	0.643	0.655	0.657	0.656	**0.538**	0.490	0.513
WLV [17]	-	-	-	-	-	-	-	-	-	0.400	0.430	0.414
Neur-bsl [29]	0.605	0.687	**0.644**	0.588	0.728	0.651	0.634	**0.676**	0.654	0.432	0.532	0.477
Stack-NER [2]	0.515	0.562	0.537	0.587	0.662	0.622	**0.750**	0.670	**0.708**	0.390	0.472	0.427
LLaMA2$_{linear}$	0.185	0.195	0.190	0.178	0.343	0.235	0.133	0.318	0.187	0.242	0.393	0.299
LIT	0.545	**0.705**	0.615	0.603	**0.770**	**0.676**	0.460	**0.723**	0.562	0.475	**0.646**	**0.548**

By comparing Stack-NER, LIT, and the remaining two methods, the inclusion of an encoder significantly improves the model's effectiveness. The addition

[6] Historical NER: `https://huggingface.co/dbmdz/bert-base-historic-multilingual-64k-td-cased`;
ATE: `https://huggingface.co/bert-base-cased`

[7] All evaluations were performed with https://github.com/hipe-eval/HIPE-scorer

of extra encoders to further process the semantic vectors output from the backbone's language model compensates for the bias that arises from using the language model directly and allows the model to perform better in the prediction phase. This further underscores the dependence on data of transformers' architectures. Despite having a sizable training corpus in English, historical training data remains insufficient. Meanwhile, introducing a decoder for prediction, this new approach is competitive in the face of traditional methods for CRFs and mapping layers. However, for the French dataset, the LIT model does not yield the best results due to the large volume of data and the fact that the benchmarks were all trained using additional auxiliary datasets.

Besides, our model demonstrates superior performance compared to the benchmark in recall across all datasets, indicating its effectiveness in classifying named entities. However, when it comes to precision, our model still falls short of the benchmark. This highlights that while our model excels at determining if a token is a named entity, it struggles to accurately identify the specific named entity among multiple tokens. Consequently, many tokens labeled as non-NE are inaccurately classified by our model. We observe that our model effectively identifies labels with consistent patterns but encounters difficulties with non-NE tokens, which encompass a wider range of entity representations. In prediction, the query in the decoder represents the embedding of a class of labels, and the final result is obtained by comparing the cosine similarity. When facing non-NEs, it is difficult to train a query that can cover all embeddings because it corresponds to a large number of tokens, which causes the phenomenon described above.

Table 3. Detailed historical NER results per entity type for the LIT system

Label	**NewsEye**									**HIPE-2020**		
	Finnish			**Swedish**			**French**			**English**		
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
fuzzy												
PER	0.801	0.912	0.853	0.685	0.947	0.795	0.635	0.924	0.750	0.753	0.936	0.834
LOC	0.633	0.897	0.742	0.795	0.942	0.863	0.640	0.878	0.740	0.677	0.845	0.752
ORG	0.385	0.556	0.455	0.436	0.567	0.493	0.348	0.853	0.490	0.374	0.645	0.473
HUMANPROD	0.442	0.864	0.585	0.576	0.864	0.691	0.351	0.818	0.491	0.240	0.316	0.273
TIME	-	-	-	-	-	-	-	-	-	0.500	1.000	0.667
strict												
PER	0.649	0.739	0.691	0.540	0.746	0.627	0.511	0.743	0.600	0.567	0.705	0.629
LOC	0.515	0.729	0.603	0.730	0.866	0.792	0.533	0.731	0.610	0.597	0.746	0.663
ORG	0.256	0.370	0.303	0.282	0.367	0.319	0.265	0.650	0.370	0.244	0.421	0.309
HUMANPROD	0.349	0.682	0.462	0.485	0.727	0.582	0.273	0.636	0.380	0.080	0.105	0.091
TIME	-	-	-	-	-	-	-	-	-	0.324	0.647	0.431

Furthermore, the analysis of historical NER per entity type shown in Table 3 reveals that our model performs significantly better at identifying entities categorized as *PER*, *TIME*, and *LOC* in comparison with *PROD* and *ORG*. The

language model faces difficulty in embedding tokens that differ substantially in semantics or representation from those found in the modern corpora, which consequently affects our model negatively. For instance, *HUMANPROD* and *ORG* which represent names of media and institutions in historical documents respectively, are difficult to locate within the modern corpus. In contrast, entities such as *PER* have similar expressions in both historical and contemporary corpora. The historical NER results on French further demonstrate the impact of utilizing a language model fine-tuned with a large-scale NER dataset, surpassing the performance of our model fine-tuned solely with a small corpus during the experiments. This reaffirms the significant influence of the language model's ability to embed historical NER corpus on our model's performance.

As seen in Table 3, our model is able to capture entities of type *PER*, *TIME*, and *LOC* very well, and their corresponding F1-score are all above 74.0 and 60.3 percent in the fuzzy and strict evaluations, respectively. In contrast, the results of *ORG* and *HUMANPROD* are poor with the best F1-score being only 69.1 percent in the fuzzy evaluation and only 58.2 percent in the strict evaluation. This is due to the fact that: (1) the amount of these two labels in the corpus is much smaller than others; (2) the language model does not produce a discriminative embedding vector. As a result, the model does not learn well the corresponding semantics, so even small prediction errors can have a significant impact on the final results during evaluation.

It is worth noting that simply using the word vectors obtained by LLaMA2 and utilizing the linear layer for classification did not yield the desired results, and this approach did not outperform traditional language model-based approaches on all datasets. There has been related work that also demonstrates that using the promotion approach, LLM falls short of traditional methods on NER tasks [19], especially for historical text [9], and does not outperform traditional models like text generation. The reason for this phenomenon is first that the training task of the LLaMA2 model is text generation rather than text understanding, which causes the generated semantics are not adapted to the classification task. Due to the cause mask of LLaMA2 model, the semantic vectors of non-end tokens cannot obtain the semantics of other tokens, which has a negative impact on the NER task. Secondly, the semantics of many words have changed due to time, but the pre-trained model lacks sufficient historical corpus for training, which also affects the quality of the generated semantic vectors. Improvements to LLaMA2's cause mask, or fine-tuning using historical text data could potentially improve its performance.

5.2 Term Extraction

We compare the performance of our proposed model with the solution of the winning teams of the TermEval competition, including the work of TALN-LS2N for English, and NLPLab_UQAM for Dutch, as well as the BERT benchmark that was pre-trained specifically for the language that we evaluate, including BERT$_{Tran22}$ for English corpus and roBERT-base$_{Tran22}$ for Dutch corpus.

The results for both gold standard versions of the English and Dutch ACTER datasets are presented in Table 4.

Table 4. Comparative results of ATE on the ACTER dataset

Models	English			Dutch		
	P	**R**	**F1**	**P**	**R**	**F1**
ANN version						
TALN-LS2N [12]	0.326	**0.727**	0.450	-	-	-
NLPLab_UQAM [24]	0.201	0.160	0.178	0.290	0.104	0.153
BERT$_{Tran22}$ [34]	**0.591**	0.324	0.419	-	-	-
roBERT-base$_{Tran22}$ [34]	-	-	-	**0.696**	0.368	0.482
LIT	0.370	0.704	**0.485**	0.498	**0.735**	**0.594**
NES version						
TALN-LS2N [12]	0.348	**0.709**	0.467	-	-	-
NLPLab_UQAM [24]	0.214	0.156	0.181	0.189	0.186	0.187
BERT$_{Tran22}$ [34]	**0.614**	0.475	**0.536**	-	-	-
roBERT-base$_{Tran22}$ [34]	-	-	-	**0.716**	0.550	**0.622**
LIT	0.450	0.651	0.532	0.500	**0.803**	0.616

Regarding the ANN version where NEs are not included in the gold standard, our additional information into BERT architectures demonstrates superior performance in F1-score in comparison with the solutions presented by the two winning teams (TALN-LS2N and NLPLab_UQAM) for both languages as well as in comparison to the original BERT designed specifically for each language. Conversely, in the NES version, where both terms and NEs are considered as the ground-truth terms, our proposed architecture significantly enhances recall performance surpassing all benchmark approaches. Furthermore, F1-score of LIT outperforms the scores achieved by the winning teams. However, it falls short of surpassing the performance of the original BERT architecture (BERT$_{Tran22}$ and roBERT-base$_{Tran22}$) for English and Dutch specifically.

Overall, the results of our proposed architecture demonstrated a discrepancy when it comes to the extraction of the ANN and NES gold standards from the ACTER corpus where it performs better for the version without NEs in the gold standard. We hypothesized that this behavior may be attributed to the dissimilarity in term length between the two annotation categories, primarily due to the inclusion of lengthy NEs.

6 Conclusion

In this paper, we propose a novel end-to-end architecture that leverages transformers' encoder-decoder framework, enhanced by the integration of semantic information similarity labels. The results demonstrate the capability of our additional semantic similarity label information when compared to the conventional

transformer-based architecture, which has been proved in two token classification tasks (i.e., historical NER and ATE) and a sequence-based classification task (i.e., sentiment analysis). In the context of token classification, our model excels at extracting the candidate terms in ATE where entities are excluded from the gold standards, and at accurately detecting candidate entities and their corresponding types in NER. However, the experiments also showed the need for our method to improve the recognition performance on no-entity in order to improve the precision values in the NER task. For all datasets and named entity kinds, the results of fuzzy validation are better than those of strict validation, which indicates that there is still a lot of room for improvement in the model to perform fully accurate named entity extraction.

In the future, we would like to investigate the substantial impact of LLMs as input representation with further exploration of innovative techniques such as prompt designing and instruction tuning. This approach aims to maximize the adaptability of our architecture to diverse NLP tasks by fine-tuning instructions and prompts to specific requirements. Moreover, we envision a deeper exploration of active learning strategies to enhance our model's learning process, enabling it to efficiently select and annotate data for training, thus reducing human annotation efforts and potentially improving model performance.

Acknowledgments. This work has been supported by the ANNA (2019-1R40226), TERMITRAD (2020-2019-8510010), Pypa (AAPR2021-2021-12263410), and Actuadata (AAPR2022-2021-17014610) projects funded by the Nouvelle-Aquitaine Region (France).

Disclosure of Interests. The authors have no competing interests to declare relevant to this article's content.

References

1. Almazrouei, E., Alobeidli, H., Alshamsi, A., Cappelli, A., Cojocaru, R., Debbah, M., Goffinet, E., Heslow, D., Launay, J., Malartic, Q., Noune, B., Pannier, B., Penedo, G.: The falcon series of language models:towards open frontier models (2023)
2. Boroş, E., Hamdi, A., Pontes, E.L., Cabrera-Diego, L.A., Moreno, J.G., Sidere, N., Doucet, A.: Alleviating digitization errors in named entity recognition for historical documents. In: Proceedings of the 24th conference on computational natural language learning. pp. 431–441 (2020)
3. Crane, G., Jones, A.: The challenge of virginia banks: an evaluation of named entity analysis in a 19th-century newspaper collection. In: Proceedings of the 6th ACM/IEEE-CS joint conference on Digital libraries. pp. 31–40 (2006)
4. Ehrmann, M., Hamdi, A., Pontes, E.L., Romanello, M., Doucet, A.: Named entity recognition and classification in historical documents: A survey. ACM Computing Surveys **56**(2), 1–47 (2023)
5. Ehrmann, M., Romanello, M., Clematide, S., Ströbel, P., Barman, R.: Language resources for historical newspapers: the impresso collection (2020)

6. Ehrmann, M., Romanello, M., Flückiger, A., Clematide, S.: Extended overview of clef hipe 2020: named entity processing on historical newspapers. In: CLEF 2020 Working Notes. Conference and Labs of the Evaluation Forum. vol. 2696. CEUR-WS (2020)
7. Ehrmann, M., Romanello, M., Najem-Meyer, S., Doucet, A., Clematide, S., Faggioli, G., Ferro, N., Hanbury, A., Potthast, M.: Extended overview of hipe-2022: Named entity recognition and linking in multilingual historical documents. In: CEUR Workshop Proceedings. pp. 1038–1063. No. 3180, CEUR-WS (2022)
8. Floridi, L., Chiriatti, M.: Gpt-3: Its nature, scope, limits, and consequences. Minds and Machines **30**, 681–694 (2020)
9. González-Gallardo, C.E., Boros, E., Girdhar, N., Hamdi, A., Moreno, J.G., Doucet, A.: Yes but.. can chatgpt identify entities in historical documents? In: 2023 ACM/IEEE Joint Conference on Digital Libraries (JCDL). pp. 184–189. IEEE (2023)
10. González-Gallardo, C.E., Tran, T.H.H., Girdhar, N., Boroş, E., Moreno, J.G., Doucet, A.: L3i++ at semeval-2023 task 2: Prompting for multilingual complex named entity recognition. In: Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023). pp. 807–814 (2023)
11. Hamdi, A., Linhares Pontes, E., Boros, E., Nguyen, T.T.H., Hackl, G., Moreno, J.G., Doucet, A.: A multilingual dataset for named entity recognition, entity linking and stance detection in historical newspapers. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 2328–2334 (2021)
12. Hazem, A., Bouhandi, M., Boudin, F., Daille, B.: TermEval 2020: TALN-LS2N System for Automatic Term Extraction. In: Proceedings of the 6th International Workshop on Computational Terminology. pp. 95–100 (2020)
13. Ivačič, N., Tran, T.H.H., Koloski, B., Pollak, S., Purver, M.: Analysis of transfer learning for named entity recognition in south-slavic languages. In: Proceedings of the 9th Workshop on Slavic Natural Language Processing 2023 (SlavicNLP 2023). pp. 106–112 (2023)
14. Karimi, A., Rossi, L., Prati, A.: Improving bert performance for aspect-based sentiment analysis. arXiv preprint arXiv:2010.11731 (2020)
15. Kenton, J.D.M.W.C., Toutanova, L.K.: Bert: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of naacL-HLT. vol. 1, p. 2 (2019)
16. Koufakou, A., Pamungkas, E.W., Basile, V., Patti, V., et al.: Hurtbert: Incorporating lexical features with bert for the detection of abusive language. In: Proceedings of the fourth workshop on online abuse and harms. pp. 34–43. Association for Computational Linguistics (2020)
17. Labusch, K., Neudecker, C.: Entity linking in multilingual newspapers and classical commentaries with bert (2022)
18. Lang, C., Wachowiak, L., Heinisch, B., Gromann, D.: Transforming term extraction: Transformer-based approaches to multilingual term extraction across domains. In: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021. pp. 3607–3620 (2021)
19. Li, Z., Li, X., Liu, Y., Xie, H., Li, J., Wang, F.l., Li, Q., Zhong, X.: Label supervised llama finetuning. arXiv preprint arXiv:2310.01208 (2023)
20. Lin, Y., Meng, Y., Sun, X., Han, Q., Kuang, K., Li, J., Wu, F.: Bertgcn: Transductive text classification by combining gnn and bert. In: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021. pp. 1456–1462 (2021)

21. Mutinda, J., Mwangi, W., Okeyo, G.: Sentiment analysis of text reviews using lexicon-enhanced bert embedding (lebert) model with convolutional neural network. Applied Sciences **13**(3), 1445 (2023)
22. OpenAI: Gpt-4 technical report (2023)
23. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. The Journal of Machine Learning Research **21**(1), 5485–5551 (2020)
24. Rigouts Terryn, A., Hoste, V., Drouin, P., Lefever, E.: TermEval 2020: Shared Task on Automatic Term Extraction Using the Annotated Corpora for Term Extraction Research (ACTER) Dataset. In: 6th International Workshop on Computational Terminology (COMPUTERM 2020). pp. 85–94. European Language Resources Association (ELRA) (2020)
25. Rigouts Terryn, A., Hoste, V., Lefever, E.: In no uncertain terms: a dataset for monolingual and multilingual automatic term extraction from comparable corpora. Language Resources and Evaluation **54**(2), 385–418 (2020)
26. Ritze, D., Zirn, C., Greenstreet, C., Eckert, K., Ponzetto, S.P.: Named entities in court: The marinelives corpus. In: Language Resources and Technologies for Processing and Linking Historical Documents and Archives-Deploying Linked Open Data in Cultural Heritage–LRT4HDA Workshop Programme. p. 26 (2014)
27. Rosset, S., Grouin, C., Zweigenbaum, P.: Entités nommées structurées: guide d'annotation Quaero. LIMSI-Centre national de la recherche scientifique (2011)
28. Ryser, A., Nguyen, Q.A., Bodenmann, N., Chen, S.Y.: Exploring transformers for multilingual historical named entity recognition (2022)
29. Schweter, S., März, L., Schmid, K., Çano, E.: hmbert: Historical multilingual language models for named entity recognition (2022)
30. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al.: Llama: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971 (2023)
31. Tran, H.T.H., Doucet, A., Sidere, N., Moreno, J.G., Pollak, S.: Named entity recognition architecture combining contextual and global. In: Towards Open and Trustworthy Digital Societies: 23rd International Conference on Asia-Pacific Digital Libraries, ICADL 2021, Virtual Event, December 1–3, 2021, Proceedings. p. 264. Springer Nature (2021)
32. Tran, H.T.H., Martinc, M., Caporusso, J., Doucet, A., Pollak, S.: The recent advances in automatic term extraction: A survey. arXiv preprint arXiv:2301.06767 (2023)
33. Tran, H.T.H., Martinc, M., Doucet, A., Pollak, S.: Can cross-domain term extraction benefit from cross-lingual transfer? In: Discovery Science: 25th International Conference, DS 2022, Montpellier, France, October 10–12, 2022, Proceedings. pp. 363–378. Springer (2022)
34. Tran, H.T.H., Martinc, M., Pelicon, A., Doucet, A., Pollak, S.: Ensembling transformers for cross-domain automatic term extraction. In: From Born-Physical to Born-Virtual: Augmenting Intelligence in Digital Libraries: 24th International Conference on Asian Digital Libraries, ICADL 2022, Hanoi, Vietnam, November 30–December 2, 2022, Proceedings. pp. 90–100. Springer (2022)
35. Tran, H.T.H., Martinc, M., Repar, A., Ljubešić, N., Doucet, A., Pollak, S.: Can cross-domain term extraction benefit from cross-lingual transfer and nested term labeling? Machine Learning **113**(7), 4285–4314 (2024)
36. Tran, H., Martinc, M., Doucet, A., Pollak, S.: A transformer-based sequence-labeling approach to the slovenian cross-domain automatic term extraction. In: Slovenian conference on Language Technologies and Digital Humanities (2022)

37. Won, M., Murrieta-Flores, P., Martins, B.: ensemble named entity recognition (ner): evaluating ner tools in the identification of place names in historical corpora. Frontiers in Digital Humanities **5**, 2 (2018)
38. Xu, Y., Li, M., Cui, L., Huang, S., Wei, F., Zhou, M.: Layoutlm: Pre-training of text and layout for document image understanding. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 1192–1200 (2020)
39. Yang, Y., Katiyar, A.: Simple and effective few-shot named entity recognition with structured nearest neighbor learning. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 6365–6375 (2020)
40. Yang, Z., Chen, H., Zhang, J., Ma, J., Chang, Y.: Attention-based multi-level feature fusion for named entity recognition. In: Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence. pp. 3594–3600 (2021)
41. Yao, L., Mao, C., Luo, Y.: Kg-bert: Bert for knowledge graph completion. arXiv preprint arXiv:1909.03193 (2019)
42. Yu, S., Su, J., Luo, D.: Improving bert-based text classification with auxiliary sentence and domain knowledge. IEEE Access **7**, 176600–176612 (2019)
43. Zhang, Z., Han, X., Liu, Z., Jiang, X., Sun, M., Liu, Q.: Ernie: Enhanced language representation with informative entities. arXiv preprint arXiv:1905.07129 (2019)