Contextual Probing for Low-Resource Named Entity Recognition in Latin

Maria Mihaela Trusca, Mark Depauw, Violet Soen,
Ine de Daele, Kevin Verbruggen, Tim Van de Cruys
Faculty of Arts, KU Leuven
Blijde Inkomststraat 21, Leuven, 3000, Belgium
mariamihaela.trusca@kuleuven.be, mark.depauw@kuleuven.be, violet.soen@kuleuven.be,
ine.dedaele@kuleuven.be, kevin.verbruggen@kuleuven.be, tim.vandecruys@kuleuven.be

Abstract

Named Entity Recognition (NER) for low-resource languages remains challenging due to limited annotated data and linguistic characteristics such as rich morphology and flexible word order. In this work, we propose a probing-based method that leverages the contextual knowledge encoded in pretrained language models to detect entities. Our approach uses a substitution strategy in which words in a sentence are replaced, one by one, with candidate entities of predefined entity types, referred to as probes. By measuring how well the probes of a certain entity type fit the surrounding context of the replaced word, we estimate the compatibility between the replaced word and the entity type. The resulting compatibility scores can be used either as a standalone zero-shot NER model or as an auxiliary feature during NER model decoding. We evaluate our method on the Latin dataset provided in the Fourth Workshop on Language Technologies for Historical and Ancient Languages (LT4HALA). Our system ranked second in the coarse-grained NER task. For the fine-grained NER task, where no training data were available, we relied exclusively on the proposed scoring method without any model training and achieved third place. These results demonstrate that contextual probing can provide an effective signal for NER in low-resource settings.

Keywords: Contextual Probing, Latin Named Entity Recognition, Zero-Shot Named Entity Recognition

1. Introduction

Named Entity Recognition (NER) is a core task in Natural Language Processing (NLP), with applications in question answering, information retrieval, and digital humanities research. While NER models have achieved strong performance for high-resource languages such as English, their application to low-resource languages, such as Latin, remains limited. This is partly due to linguistic factors such as rich morphology and flexible word order, which are difficult to model effectively when annotated training data are limited.

The current NER standard approach relies on pre-trained language models for the target language, which are subsequently fine-tuned for NER. Although this strategy is generally robust, the final performance of the NER system largely depends on the size and quality of the available training data.

In this paper, we propose a method designed to reduce the negative impact of limited NER data by more effectively leveraging the linguistic knowledge acquired by the language models during their pretraining. This knowledge is learned through a self-supervised language modeling task, in which the model is trained to predict missing or contextually appropriate tokens, encouraging it to capture rich contextual relationships between words. In our implementation, we exploit these contextual relationships to evaluate how well substitutions of well-known entities fit in the surrounding context, allowing us to infer whether a given position in the text likely corresponds to an entity.

The core idea of our method is simple yet effective. Given a predefined list of entities (or probes) of a specific type, we use a substitution-based probing approach to identify entities. For each word in the text, we replace it individually with each probe from a predefined list of known entities of a given type. When the language model predicts that the substitution fits naturally in context, this indicates that the original word is functionally similar to the probe, and we therefore infer that it is an entity of that type.

The proposed method does not require additional training and is suitable for zero-shot NER. However, when NER training data are available, the compatibility between probes and the context can be used as a complementary signal to better guide NER models during decoding in the fine-tuning process. We demonstrate the utility of our approach by evaluating it on Latin, a low-resource language, using the test dataset provided by the Fourth Workshop on Language Technologies for Historical and Ancient Languages (LT4HALA).

The rest of the paper is organized as follows. Section 2 reviews related work. Section 3 presents our method. Section 4 describes the experimental setup and Section 5 reports the results. Section 6 concludes the paper.

Proceedings of LT4HALA 2026 @ LREC 2026, pages 437–442.

2. Related Works

2.1. Probing for Named Entity Recognition

Recent work has explored probing and few-shot techniques to improve NER performance in low-resource settings. Jiang et al. (2024) propose point in-context learning (P-ICL), where lists of representative entities are provided as few-shot prompts to a language model. Cocchieri et al. (2025) instead leverage natural language definitions of entity types, assigning NER labels based on the similarity between work or span embeddings and entity type embeddings. Shachar et al. (2025) use a similar embedding-based similarity but reformulate NER as a retrieval task, ranking candidate entities rather than producing NER labels. Zhang et al. (2023) and Liang et al. (2024) combine the strengths of both approaches by constructing prototypes that merge embeddings of example entities or probes with embeddings of type definitions, and assign NER labels to tokens based on their similarity to these prototypes.

While the above models use probing either with prompting or focusing on similarity between candidate entities and type definition, in our work, probing is used to leverage the sentence context, which serves as the primary signal for determining NER labels.

2.2. Latin Named Entity Recognition

Research on Latin NER has explored a variety of approaches, ranging from traditional sequence-labelling techniques such as CRF and rule-based methods (Erdmann et al., 2016; Chastang et al., 2021; Burns, 2023; Novotny et al., 2023) to multilingual or Latin-pretrained transformer-based models (Torres Aguilar, 2022; Beersmans et al., 2023). Building on this trend, in this work, we define NER models based on Latin-pretrained transformers, since pretraining on the target language allows the model to better capture the morphology and syntax of that language than multilingual models, especially when sufficient monolingual data is available (Rust et al., 2020).

3. Methodology

3.1. Preliminaries

Pseudo-log-likelihood for Word Scoring To compute the compatibility between a word and its context, we use the Pseudo-Log-Likelihood (PLL) metric defined by Salazar et al. (2020). Given a sentence s and a word at position i, y_i, with the surrounding context $y_{\setminus i}$, the PLL is defined as

$$\text{PLL}(y_i \mid y_{\setminus i}) = -\log P(y_i \mid y_{\setminus i}). \quad (1)$$

Probes as Candidate Entities Given that our NER task involves K entity types (e.g., “place” or “person”), we define, for each type k ($k \in K$), a set of m entities, referred to as probes. To evaluate the compatibility of these probes with the sentences, it is important to select well-known or representative entities.

3.2. PLL-Based Entity Compatibility Scoring

Our hypothesis is that language models naturally encode NER-like knowledge due to their pretraining on a language modeling objective, which facilitates the learning of rich contextual representations between words. Consequently, even without any NER fine-tuning, a pretrained language model can already provide an indication of whether a word is likely to be an entity based on its surrounding context.

Problem Definition. Let s be a sentence composed of n words $(y_1, \ldots, y_n)$, and let $\mathcal{E}_k$ denote a set of m probe entities associated with entity type k. Our goal is to assign a compatibility score to each word position i for each entity type k, indicating whether the word y_i behaves like an entity of that type.

Contextual Compatibility Scoring For each word position i, we compute the compatibility score $comp_i(k)$ with respect to entity type k using PLL scores between the surrounding context of position i, denoted $y_{\setminus i}$, and the probe entities in $\mathcal{E}_k$:

$$comp_i(k) = \frac{1}{m} \sum_{e \in \mathcal{E}_k} \text{PLL}(e \mid y_{\setminus i}) \quad (2)$$

Here, $\text{PLL}(e \mid y_{\setminus i})$ measures how well the substitution of y_i with the probe entity e fits the sentence according to the pretrained language model, exploiting the contextual representations induced during pretraining.

After computing the scores $comp_i(k)$ for all K entity types, each word position i is associated with K compatibility scores, indicating its affinity to each entity type. Finally, we normalize these scores to prevent differences in scale across entity types from biasing the NER decision.

3.3. Integration with NER Model

The contextual compatibility scoring presented above has the advantage of being able to operate as an independent NER system without any fine-tuning. An example of this scoring method is shown in Table 1 for the entity types “person" and “place". While the method correctly identifies entities such as “Cicero" as a person and “Pompeiumm" as a

location, it can also make errors, as in the case of the word “habuit" Therefore, if annotated NER training data are available, we recommend integrating this scoring as an auxiliary signal during the fine-tuning of language models for NER. Specifically, given the logits $logits_i(k)$ computed by the language model for word y_i and entity type k during NER training, we adjust these logits based on the contextual compatibility scores as follows:

$$logits_i(k) = \begin{cases} logits_i(k) + \alpha \cdot comp_i(k), \\ \quad \text{if } softmax(logits_i(k)) < \tau, \\ logits_i(k), \text{otherwise.} \end{cases} \tag{3}$$

Word	Person	Place
Cicero	0.89	0.44
in	0	0
Roma	0.28	0.81
orationem	0.87	0.58
habuit	0.84	0.66
et	0	0
Pompeium	0.70	0.73
salutavit	0.1	0.04

Table 1: Contextual Compatibility Scoring Method used as a NER model for the sentence: “Cicero in Roma orationem habuit et Pompeium salutavit".

Here, α controls the strength of the contextual compatibility scoring, and τ is a threshold that ensures the scoring is applied only in uncertain or difficult cases, i.e., when the NER model is not confident about the entity type of a word. Since the NER knowledge learned from a specialized training dataset is typically more reliable than relying solely on contextual similarity learned during the pretraining of the language models, the threshold τ prevents the compatibility scores from overriding the model’s confident predictions.

4. Experimental Setup

Label	[1]	[2]	[3]	[4]	[5]
O	90.45	92.05	88.59	98.24	92.42
“person"	5.74	5.19	6.42	1.69	3.05
“place"	3.63	1.79	1.50	0.07	2.93
“collective"	0.00	0.97	3.49	0.00	1.25
“work"	0.00	0.00	0.00	0.00	0.26
“time"	0.00	0.00	0.00	0.00	0.09
“object"	0.00	0.00	0.00	0.00	0.01

Table 2: Percentage distribution of NER labels across the five datasets used for the training of the NER model.

Datasets To train the NER models, we rely on five datasets whose distribution of NER labels is presented in Table 2. The first dataset [1] was introduced by Torres Aguilar (2022) and contains Latin NER annotations for manuscripts written between the 10th and 15th centuries. The second dataset [2], presented by Beersmans et al. (2023), contains annotations for Latin texts written between the 2nd century BCE and the 2nd century CE. The third dataset [3], described by Erdmann et al. (2016), consists of Latin texts written between the 1st century BCE and the early 2nd century CE. The fourth dataset [4] is an internal resource that will soon be released as part of our work at STUDIUM.AI. It contains NER annotations for Latin student notes written between the 16th and 18th centuries.

Since the first four datasets described above contain annotations for only three of the ten NER categories defined in the LT4HALA evaluation (“person",“place", and “collective"), we extend the set of available NER labels by creating a fifth dataset [5]. This dataset is generated by translating existing HIPE datasets (Ehrmann et al., 2022) in French, English, and German into Latin using the Google Translate API. To propagate the NER labels from the source texts to the Latin translations, we first extract the entities in the source language—hereafter referred to as source entities—and then apply the following two-step alignment procedure:

- First, we translate the source entities separately from the texts into Latin. The Latin NER labels of the first step are obtained by matching the translated entities to words in the translated Latin texts.
- In the second step, for each unaligned source entity from the first step and each word in the translated Latin text, we compute the alignment score below:

$$align = \beta \cdot \cos_{sim}(emb_{src}, emb_{Latin}) + (1 - \beta) \cdot Lev_{dist}(w_{src}, w_{Latin}) \tag{4}$$

  where:

  - emb_{src} is the language-agnostic BERT (LaBERT) embedding (Feng et al., 2022) of the remaining unaligned source entity,
  - emb_{Latin} is the LaBERT embedding of a candidate word in the translated Latin text,
  - $\cos_{sim}$ is the cosine similarity between embeddings,
  - Lev_{dist} is the Levenshtein distance between the remaining unaligned source entity w_{src} and the sentence word w_{Latin}.
  - β is a weighting factor balancing semantic similarity and orthographic similarity.

Here, $\cos_{\sim}$ measures the semantic similarity, while Lev_{dist} measures the orthographic similarity. Each

remaining unaligned source entity from the first step is assigned to the word in the Latin translation with the highest $score$ exceeding the threshold ϕ. These assignments constitute the Latin NER labels produced in the second step of our procedure. The final NER labels for the translated Latin texts are obtained by merging the labels from both steps of the alignment process.

Implementation details All our experiments are performed on an NVIDIA GeForce RTX 3090 GPU. The language models are trained for 600 epochs, with a learning rate of 5×10^{-5} and a batch size of 16. The hyperparameters of our method are set as follows: $\alpha = 1$, $\tau = 0.7$, $\beta = 0.9$, and $\phi = 0.97$.

Models To evaluate our PLL-based entity compatibility scoring, we use LaBERTa (Riemenschneider and Frank, 2023), a RoBERTa-based (Liu et al., 2019) encoder pretrained on Latin corpora. In Section 5 dedicated to the experiments, our LaBERTa-based NER model with the proposed PLL method is referred to as "KULeuven," which is the name of our team for the LT4HALA competition. For comparison, we also report results using LatinBERT (Bamman and Burns, 2020). Both LaBERTa and LatinBERT are encoder-only transformers suitable for token-level tasks such as NER. Preliminary experiments showed that LaBERTa outperforms LatinBERT (79.45% vs. 74.24% F1 on a 90/10 train/validation split, both integrated with PLL scoring), so only the LaBERTa-based results were submitted to the LT4HALA competition.

Metrics The NER results are evaluated using precision, recall, and F1 scores under two settings: *strict* and *fuzzy*. The *strict* setting requires an exact match of both entity type and boundaries, while the *fuzzy* setting requires the correct entity type with at least one work overlap.

5. Experiments

5.1. LT4HALA Tasks

We evaluate our model on the two shared tasks of the LT4HALA competition.

Task 1 – NERC Coarse-grained: Recognize top-level entity categories (i.e., entities like "person" and "place"). Since annotated NER training data is available, labels were generated using LaBERTa fine-tuned on the data and combined with our PLL-based entity compatibility scoring. Our model, referred to as "KULeuven", the name of our team for the LT4HALA competition, ranked second among competitors (Table 3).

Task 2 – NERC Fine-grained: Recognize top- and sub-level entity categories (e.g., "person.entity", "person.ancesty", or "collective.organization"). For this task, no training data is available. Instead, we define lists of probes for all pairs of top- and sub-level entities and use them to assign NER labels via the PLL-based scoring method, without integrating a language model. While our method ranks third, it is a zero-shot approach that requires no fine-tuning and relies solely on the knowledge encoded in the pretrained LaBERTa model.

Evaluation	System	P	R	F1
strict	uOttawa	0.899	0.917	0.908
	KULeuven	0.736	0.694	0.714
	argo-navis	0.575	0.498	0.534
fuzzy	uOttawa	0.932	0.950	0.941
	KULeuven	0.794	0.749	0.771
	argo-navis	0.635	0.550	0.590

Table 3: Task 1 – NERC Coarse-grained. Strict and fuzzy comparison between the results of our team (KULeuven) and the results of the other two teams (uOttawa and argo-navis). For each team, we present the best obtained results. The results of our team are obtained by integrating the proposed PLL-based method into LaBERTa.

Evaluation	System	P	R	F1
strict	uOttawa	0.841	0.890	0.865
	argo-navis	0.466	0.327	0.384
	KULeuven	0.131	0.128	0.129
fuzzy	uOttawa	0.862	0.913	0.887
	argo-navis	0.523	0.367	0.432
	KULeuven	0.141	0.138	0.140

Table 4: Task 1 – NERC Fine-grained. Strict and fuzzy comparison between the results of our team (KULeuven) and the results of the other two teams (uOttawa and argo-navis). For each team, we present the best obtained results. The NER labels generated by our model are extracted using only our PLL-based scoring method with no fine-tuning.

5.2. Effect of the PLL-based Method on NER Performance

While the results submitted to LT4HALA rely on LaBERTa due to its strong NER performance on Latin texts, this subsection examines in more detail the general impact of the proposed PLL-based method on the NER capabilities of Latin models, including not only LaBERTa but also LatinBERT.

In Table 5, we compare LaBERTa and LatinBERT integrated with our PLL-based scoring method for Task 1 of the LT4HALA competition. We also report results for the zero-shot setting, where NER labels are obtained solely using our PLL-based entity compatibility scoring method computed from the pretrained LaBERTa or LatinBERT models; these

results are marked with * in the table. Additionally, we include results for LaBERTa and LatinBERT trained on the NER data without applying the PLL-based method.

As expected, adding our PLL-based method during the decoding process of NER fine-tuning improves performance for both Latin language models. Among the evaluated setups, LaBERTa combined with PLL (referred to as "KULeuven" in Table 3) achieves the best overall performance. Notably, the zero-shot approach remains competitive, with an F1 score exceeding 0.304 under strict evaluation and 0.338 under fuzzy evaluation, despite the absence of NER fine-tuning.

Evaluation	System	P	R	F1
strict	PLL+LaBERTa	0.736	0.694	0.714
	PLL+LatinBert	0.679	0.652	0.665
	PLL+LaBERTa*	0.336	0.387	0.359
	PLL+LatinBERT*	0.285	0.325	0.304
	LaBERTa	0.658	0.631	0.644
	LatinBERT	0.612	0.624	0.618
fuzzy	PLL+LaBERTa	0.794	0.749	0.771
	PLL+LatinBert	0.713	0.684	0.697
	PLL+LaBERTa*	0.384	0.424	0.403
	PLL+LatinBERT*	0.341	0.335	0.338
	LaBERTa	0.724	0.704	0.714
	LatinBERT	0.677	0.635	0.656

Table 5: Task 1 – NERC Coarse-grained. Strict and fuzzy NER performance of LaBERTa and LatinBERT with PLL. PLL+LaBERTa is refered as "KULeuven" in Table 3. Results marked with * correspond to the zero-shot setting with no fine-tuning. Additionally, we report results for the fine-tuned LaBERTa and LatinBERT on the NER data without the PLL-based method.

In Table 6, we compare our PLL-based scoring method applied in a zero-shot framework on top of LaBERTa and LatinBERT for Task 2 of the LT4HALA competition. The PLL-based method is applied as a standalone NER model due to the lack of NER training data for this task. Despite the large number of probes used to differentiate all pairs of top- and sub-level entity categories and the absence of training data, the PLL-based method produces competitive results on both models, with LaBERTa performing slightly better.

Evaluation	System	P	R	F1
strict	PLL+LaBERTa*	0.131	0.128	0.129
	PLL+LatinBERT*	0.109	0.102	0.105
fuzzy	PLL+LaBERTa*	0.141	0.138	0.140
	PLL+LatinBERT*	0.135	0.118	0.126

Table 6: Task 1 – NERC Fine-grained. Strict and fuzzy NER performance of LaBERTa and LatinBERT with PLL in the zero-shot setting (*). PLL+LaBERTa is refered as "KULeuven" in Table 4.

6. Conclusion

We introduce a novel approach to better exploit the implicit NER knowledge acquired by language models during pretraining and demonstrate its effectiveness for Latin. Our method can be integrated into NER models to enhance fine-tuning, or used as a standalone zero-shot approach when no annotated data is available. It achieved second place in coarse-grained NER and third in fine-grained NER at the LT4HALA workshop, highlighting that pretrained models encode rich entity knowledge that can be leveraged effectively in low-resource NER scenarios.

7. Acknowledgements

This work is part of the Studium.AI project funded by FW0 I004022N.

8. Bibliographical References

David Bamman and Patrick J. Burns. 2020. Latin bert: A contextual language model for classical philology. *ArXiv*, abs/2009.10053.

Marijke Beersmans, Evelien de Graaf, Tim Van de Cruys, and Margherita Fantoli. 2023. Training and evaluation of named entity recognition models for classical Latin. In *Proceedings of the Ancient Language Processing Workshop*, pages 1–12. INCOMA Ltd., Shoumen, Bulgaria.

Patrick J. Burns. 2023. Latincy: Synthetic trained pipelines for latin nlp. *ArXiv*, abs/2305.04365.

Pierre Chastang, Sergio Torres Aguilar, and Xavier Tannier. 2021. A named entity recognition model for medieval latin charters. *Digit. Humanit. Q.*, 15.

Alessio Cocchieri, Marcos Martínez Galindo, Giacomo Frisoni, Gianluca Moro, Claudio Sartori, and Giuseppe Tagliavini. 2025. Zeroner: Fueling zero-shot named entity recognition via entity type descriptions. In *Annual Meeting of the Association for Computational Linguistics*.

Maud Ehrmann, Matteo Romanello, Antoine Doucet, and Simon Clematide. 2022. Introducing the hipe 2022 shared task: Named entity recognition and linking in multilingual historical documents. In *Advances in Information Retrieval*, pages 347–354. Springer International Publishing.

Alexander Erdmann, Christopher Brown, Brian Joseph, Mark Janse, Petra Ajaka, Micha Elsner,

and Marie-Catherine de Marneffe. 2016. Challenges and solutions for Latin named entity recognition. In *Proceedings of the Workshop on Language Technology Resources and Tools for Digital Humanities (LT4DH)*, pages 85–93. The COLING 2016 Organizing Committee.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891. Association for Computational Linguistics.

Guochao Jiang, Zepeng Ding, Yuchen Shi, and Deqing Yang. 2024. P-icl: Point in-context learning for named entity recognition with large language models. *ArXiv*, abs/2405.04960.

Yueqing Liang, Liangwei Yang, Chen Wang, Xiongxiao Xu, Philip S. Yu, and Kai Shu. 2024. Taxonomy-guided zero-shot recommendations with llms. *ArXiv*, abs/2406.14043.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692.

Vit Novotny, Kristina Luger, Michal Štefánik, Tereza Vrabcova, and Ales Horak. 2023. People and places of historical Europe: Bootstrapping annotation pipeline and a new corpus of named entities in late medieval texts. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 14104–14113. Association for Computational Linguistics.

Frederick Riemenschneider and Anette Frank. 2023. Exploring large language models for classical philology. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL'23)*. Association for Computational Linguistics, Toronto, Canada.

Phillip Rust, Jonas Pfeiffer, Ivan Vulic, Sebastian Ruder, and Iryna Gurevych. 2020. How good is your tokenizer? on the monolingual performance of multilingual language models. *ArXiv*, abs/2012.15613.

Julian Salazar, Davis Liang, Toan Q. Nguyen, and Katrin Kirchhoff. 2020. Masked language model scoring. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 2699–2712. Association for Computational Linguistics.

Or Shachar, Uri Katz, Yoav Goldberg, and Oren Glickman. 2025. Ner retriever: Zero-shot named entity retrieval with type-aware embeddings. In *Conference on Empirical Methods in Natural Language Processing*.

Sergio Torres Aguilar. 2022. Multilingual named entity recognition for medieval charters using stacked embeddings and bert-based models. In *Proceedings of the Second Workshop on Language Technologies for Historical and Ancient Languages*, pages 119–128. European Language Resources Association.

Mozhi Zhang, Hang Yan, Yaqian Zhou, and Xipeng Qiu. 2023. Promptner: A prompting method for few-shot named entity recognition via k nearest neighbor search. *ArXiv*, abs/2305.12217.