Comparative evaluation of boundary-relaxed annotation for Entity Linking performance

Gabriel Herman Bernardim Andrade **Shuntaro Yada** **Eiji Aramaki**
Graduate School of Information Science
Nara Institute of Science and Technology
Ikoma, Nara, Japan
{herman_bernardim_andrade.hi1, s-yada, aramaki}@is.naist.jp

Abstract

Entity Linking performance has a strong reliance on having a large quantity of high-quality annotated training data available. Yet, manual annotation of named entities, especially their boundaries, is ambiguous, error-prone, and raises many inconsistencies between annotators. While imprecise boundary annotation can degrade a model's performance, there are applications where accurate extraction of entities' surface form is not necessary. For those cases, a lenient annotation guideline could relieve the annotators' workload and speed up the process. This paper presents a case study designed to verify the feasibility of such annotation process and evaluate the impact of boundary-relaxed annotation in an Entity Linking pipeline. We first generate a set of noisy versions of the widely used AIDA CoNLL-YAGO dataset by expanding the boundaries subsets of annotated entity mentions and then train three Entity Linking models on this data and evaluate the relative impact of imprecise annotation on entity recognition and disambiguation performances. We demonstrate that the magnitude of effects caused by noise in the Named Entity Recognition phase is dependent on both model complexity and noise ratio, while Entity Disambiguation components are susceptible to entity boundary imprecision due to strong vocabulary dependency.

1 Introduction

Out of many tasks under the Natural Language Processing (NLP) umbrella, Entity Linking (EL) is one of the critical steps for Information Extraction, allowing the retrieval and understanding of information from unstructured textual sources. By definition, EL is the task of associating a Named Entity (NE) mention (a concept, person, location, etc.) to a unique identifier that describes what that mention refers to in a knowledge base (KB) (Zhang et al., 2021). This is especially significant in the biomedical domain, where it can be used to associate mentions with different surface forms to a single concept in an ontology, such as the Unified Medical Language System (UMLS)[1], allowing for better indexing and relation extraction (Leaman et al., 2015).

In this paper's definition, EL comprises two phases, as presented in Figure 1. Named Entity Recognition (NER) searches the sentence for NE mentions, and then Entity Disambiguation (ED) assigns a unique identifier for each one. (Kolitsas et al., 2018; Özge Sevgili et al., 2022). NER is also commonly related to classifying a detected mention into a set of categories. However, for this study, we consider it as only the location of a NE in the text.

Although some methodologies have presented state-of-the-art (SoTA) results for this task (De Cao et al., 2021; Yamada et al., 2020), large amounts of high-quality labeled data are still crucial to achieve good performance models.

Yet, text annotation is no trivial task. Manual annotation of vast data is costly due to the excessive workforce and long hours required. Besides, it is arduous to guarantee good quality and standardized annotation. Notably, when annotating NEs, annotators can face trouble deciding on the boundaries of a mention, which can be ambiguous and raises many inconsistencies (Marrero et al., 2013).

For example, in the sentence presented in Figure 1, *Arthritis* can be referred to as "*joint inflammation*". However, a longer span "*joint inflammation pain*" is also applicable when referring to the same manifestation, even though it packs some peripheral information. Deciding where the boundary line of entity or non-entity should be drawn is not straightforward, and rigorous annotation guidelines can add even more bureaucracy to an already slow process. (Chapman et al., 2011).

Although imprecise boundary annotation can be considered a type of noise that degrades a model's

[1] https://www.nlm.nih.gov/research/umls/index.html

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 8238–8253

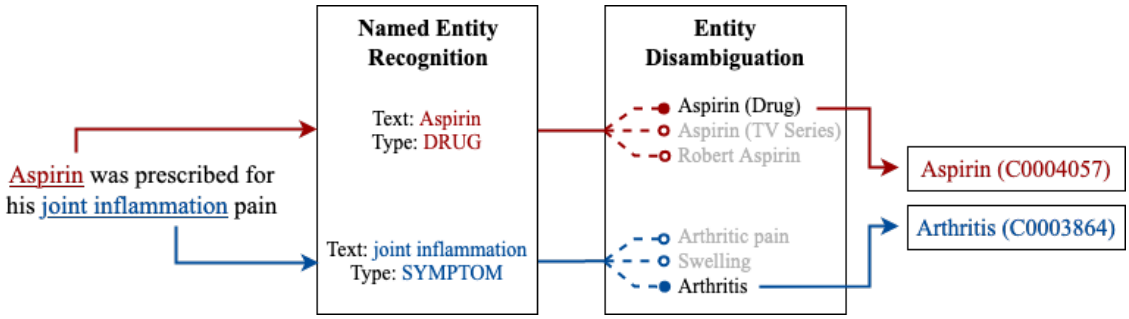

Figure 1: Example of an EL pipeline applied to a sentence. NE mentions are detected and classified by the NER stage. The ED module generates a list of candidates and ranks them to select the proper concept unique identifier.

performance, some applications, especially those in the biomedical field, are not dependent on the exact span of an entity (Tsai et al., 2006; Dai et al., 2014). Due to the existence of normalized vocabularies, most extracted data can be refined to helpful information by linking to such KBs. In this scenario, the burden on annotators could be relieved by loosening the strictness of entity boundary annotation and, instead, focusing on the complete annotation of the conceptual information.

We hypothesize that as long as the detected mention is longer than the target mention, in other words, it contains "more information than necessary", the concept matching to a KB should not be greatly degraded. This approach can potentially minimize the effort during corpora annotation process and improve its speed.

To validate the hypothesis, we present a case study to evaluate the impact of boundary-relaxed annotation on NER and ED performance. We generated multiple variants of a popular EL dataset (AIDA CoNLL-YAGO) by expanding subsets of annotated entities' boundaries. We then train two SoTA models and our baseline model on the modified dataset and compare it with the original dataset performance on each step of the EL pipeline.

2 Related Work

Entity boundary is a common theme in NER-related research. Some evaluation criteria that relax the strictness of boundary matching have been proposed and refined through the years. Tsai et al. (2006) compares a few of those strategies, such as matching only the left or right border, any tag overlap, per-token measurements, and semantically-driven matching. The authors also present potential use cases for each evaluation technique and compare their characteristics.

While several studies evaluate the impact of annotation quality in EL (Dojchinovski et al., 2016; Chen et al., 2018; Weichselbraun et al., 2019), others tackle the issue of noisy-labeled data (Liu et al., 2021; Zhu et al., 2022), which may include but it is not limited to imprecise entity boundary.

Most of the works that address the NE boundary problems attempt to improve the detection precision (Li et al., 2021; Yongming et al., 2022; Tang et al., 2022), proving that better entity boundary detection dramatically improves both entity classification and linking.

However, there are a handful of studies that specifically address the impact of relaxed entity boundary annotation on linking performance. Shmanina et al. (2013) compared the impact of two different annotation guidelines for disease names on model performance, noting that the set of rules that provide low lexical variability, short entity length, and high regularity usually impact performance positively.

As a sideline experiment to the main study, Choi and Cardie (2008) analyzed the impact of expanded entity boundaries and the use of the additional information as context around the annotated expression.

Ghiasvand and Kate (2018) developed an unsupervised method for training clinical NER systems that automatically generate NE annotations. Due to the high surface form variability of clinical concepts, a NE boundary determination component was developed. With the ablation of this component, the authors perceived around a 5% decrease in NER precision and recall performance.

Zhu and Li (2022) propose a boundary regularization technique that reallocates part of the probability to be an entity from an annotated span to its neighbor words. This effectively creates a smooth transition between an entity annotation and its non-entity surroundings to mitigate annotation boundary inconsistencies.

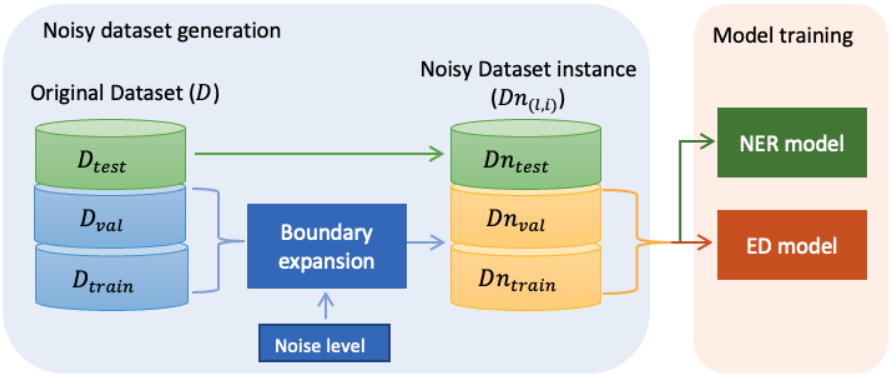

Figure 2: Workflow of our experiment.

3 Method

We performed a case study to assess the effect of relaxed annotation on the performance of each phase of the EL pipeline. Figure 2 depicts a representation of our workflow.

The designed method aims to simulate model training using a dataset compiled by employing a hypothetical lenient and inclusive entity annotation guideline, where an annotator would roughly select the correct span of the entity mentions, possibly introducing a few unnecessary extra tokens into the annotation when in uncertainty.

In particular, we focused on observing the impact of *expanded entity annotations* on evaluation metrics commonly used in such tasks.

3.1 Boundary Expansion

For that, we create a boundary-expanded version, denoted Dn, of our original dataset D, by adding tokens to some gold standard annotations. We call this processed variant as *"noisy dataset"*.

Modeling annotator mistakes in order to generate "human-like" annotation noise is not a simple task, as the type of errors committed are dependent on multiple factors such as annotator experience or on how the annotation process is conducted. For instance, the annotation interface used (by clicking/dragging or keyboard-based) can produce considerably distinct anomalies. Therefore, we developed an approach for noise generation based on randomly selecting annotations to be expanded. Although this approach may not produce natural mistakes in some situations due to word meaning not being considered upon selection, it is a simple way to avoid bias regarding the entity type, surface form, or textual position when selecting terms to be altered.

We randomly select a subset of the entity annotations from D_{train} and D_{val} and expand their boundaries by one token (word or punctuation) to the right and/or left of the original entity span, as shown in Figure 3, thus creating Dn_{train} and Dn_{val}. In order to profile the model behavior trend, we produced multiple variants of Dn with various amounts of boundary-expanded mentions. This was done by setting a percentage of mentions to be selected during the dataset processing, defined as l, where $l \in \{10, 20, 30, ..., 100\}$. We refer to l as "noise level".

Figure 3: Example of boundary expansion in an NE mention, underscored in blue. A token from either side (or both) can be selected to be merged in the annotation.

When an entity mention is selected for the subset, the expansion direction is also randomly decided. In this case, all three direction options (left, right, and both sides) were given the same probability of being chosen (1/3). Table 1 shows an example document before and after expansion.

We do not stop expanding the boundary even if there is an overlap with the next mention. In this case, both mentions are merged in a single annotation. The amount of entities that were unified for each noise level is presented in Appendix A.1.

Given that the affected mentions are randomly selected, we generate ten different instances $Dn_{(l,i)}$ for each noise level aiming to remove any bias that could be caused by the boundary expansion. We identify each instance by a i in the notation. In total, 100 different instances of Dn were produced.

Lastly, for the sake of simplicity, we limit the

Original annotation	Noisy annotation
CRICKET **- LEICESTERSHIRE TAKE** OVER AT TOP AFTER INNINGS VICTORY. **LONDON** 1996-08-30 **West Indian** all-rounder **Phil Simmons** took four for 38 on Friday as **Leicestershire** beat **Somerset** by an innings and 39 runs in two days to take over at the head of the county championship.	CRICKET - **LEICESTERSHIRE TAKE** OVER AT TOP AFTER INNINGS VICTORY. **LONDON** 1996-08-30 **West Indian all-rounder Phil Simmons** took four for 38 on Friday as **Leicestershire** beat **Somerset by** an innings and 39 runs in two days to take over at the head of the county championship.

Table 1: Comparison between the original annotation and boundary expanded annotation. The entity span is marked with bold and underscore.

noise generation length to a maximum of one-token length for each side of the NE mention. With the noise length limited, we can exclude one variable that may affect the models' performance. Instead, we focus on how the *quantity* of lenient-annotated entities influences the pipeline's output.

We then train NER and ED models on each $Dn_{(l,i)}$ and use the unmodified test set to assess the boundary relaxation leverage in performance.

3.2 Evaluation metrics

We evaluate both NER and ED models using the measurements of *precision* (fraction of extracted NEs that are correct), *recall* (fraction of NEs extracted out of all the gold-standard NEs), and *F-score* (harmonic mean of precision and recall) (Japkowicz and Shah, 2011). We use micro-averaged metrics, meaning the scores are computed across all documents.

Being aware that the NER system's capability to precisely detect entity boundaries could be affected by the noisy data training, for NER components only, two evaluation scoring schemes were used – *strict* and *relaxed*.

Strict scoring scheme follows the convention defined by CoNLL (Tjong Kim Sang and De Meulder, 2003). It only considers correct matches where boundaries exactly match the ground truth. Hence, a system gets zero credit if the extracted entity has any extra or is missing tokens compared to the gold-standard mention.

On the other hand, the *relaxed* scheme accepts mention detections with imprecise boundaries (Uzuner et al., 2011; Ghiasvand and Kate, 2018). It also considers as correct both partial matching and extra tokens in a mention, as long as the match has at least one token that overlaps with the span of the gold-standard mention.

We report NER metrics by evaluating the output of this component before being fed to the ED module of the pipeline.

The results for the ED component are calculated based on the results obtained at the end of the pipeline, in other words, the final link produced by the disambiguation model after receiving the mentions detected by the NER phase. In this case, we only use *strict* metrics, denoting that the output prediction must be the same as the gold-standard concept to be considered correct. Although a *relaxed* metric could also be employed by evaluating the KB's relationship tree between the concepts, such evaluation falls out of the scope of this study as it may mask the performance impact caused by the NER phase performance.

All reported metrics are averaged for all runs on each instance i of $Dn_{(l)}$ (noisy dataset on the referring noise level l). Therefore, an average of ten runs per noise level.

3.3 NER mention matching types

While the analysis based exclusively on the performance metrics scores allows us to grasp the trend of boundary relaxation effect in model output quality, these metrics mask distinct kinds of NER mistakes behind a single value.

We classify mention detection in six matching types and report their output percentage on each evaluated model. An example sentence for each matching type is shown in Appendix A.2).

Exact match Both detected mention boundaries match the NE mention boundaries.

Exceeding match The detected mention contains all the NE mention text and also unnecessary neighbor words.

Partial match NE mention is not completely detected, but at least a fragment was found.

Invalid match The NER component extracts a span of text where there is no real NE.

Missing match The NER component detects no part of an existing NE mention.

4 Experimental Setup

Dataset As our dataset, we use AIDA CoNLL-YAGO (Hoffart et al., 2011). This dataset is an extension of the well-known CoNLL-2003 dataset (Tjong Kim Sang and De Meulder, 2003), and it is vastly used for EL benchmarks. It comprises newswire articles from Reuters, containing 1,393 articles with 27,820 NE mentions linkable to its corresponding YAGO2 ID and Wikipedia article webpage. Training and validation sets contain 946 and 216 articles, respectively. Test set has 231 articles.

Models We executed our experimental procedure in three EL pipelines. We selected two SoTA approaches and also developed our own method:

(1) *Highly Parallel Autoregressive Entity Linking with Discriminative Correction (HPAELDC)* (De Cao et al., 2021)[2]:

An end-to-end model for autoregressive entity linking available under *MIT license*. It comprises a Longformer encoder, a mention detection module including feed-forward Neural Networks, and an ED component based on a Long Short-Term Memory network for candidate generation and ranking. The whole model has a total of 202M parameters. As reported by the authors, this model achieves an F-score of 0.85 for Entity Linking on the AIDA CoNLL-YAGO dataset. Even though the authors present it as an end-to-end model, the NER and ED modules are separated components. Therefore, we could compute metrics for each step individually.

(2) *Language Understanding with Knowledge-based Embeddings (LUKE)* (Yamada et al., 2020)[3]:

LUKE is a contextualized representation of words and entities based on a transformer. This model can perform multiple tasks, such as NER, relation classification, or question-answering. It is available under the *Apache License 2.0*.

While available in multiple variants, in this study we use *LUKE (base)*, which has a total of 253M parameters. In the NER task it achieves F-score of 0.94, and 0.95 for ED on the AIDA CoNLL-YAGO dataset. Since this framework is composed by modules that accomplish different tasks, we can independently create an EL pipeline using the available NER and ED components.

As reported in the original paper (Yamada et al., 2022), one of LUKE's ED model limitations is not being able to handle out-of-vocabulary entities, we replace expanded entities in the model's vocabulary file before training. In case of an entity being expanded differently in two or more instances, we use the shorter of all variants.

(3) *VanillaNER* [4]:

Our developed NER model is intended to be used as a baseline for comparison. This model aims to simulate a simple and low-effort approach. We start from a pre-trained *bert-base-cased* model (Devlin et al., 2019), which has around 110M parameters, and fine-tune it using our dataset instances. The base model is publicly available in the HuggingFace[5] platform, under *Apache License 2.0*. As we only developed the NER component, we feed the output of our model to LUKE's ED model, trained on the same $Dn_{(l,i)}$ instance, to compute disambiguation scores.

Training Parameters We used a server comprised of two NVIDIA Quadro RTX 8000 GPUs for model training. We report all used parameters and computational time needed in Appendix A.3.

For both SoTA models (1 and 2), we use the implementation released by the authors and also apply the default training parameters disclosed in the referring papers.

5 Results and Discussion

In this section, we present the experimental results of our case study. Appendix A.4 presents a summary of the computed evaluation metrics after the processing of the test set by the trained models in NER and ED phases, respectively. We purposefully omit some noise levels for easier visualization, presenting some checkpoints to allow comparison.

5.1 NER performance analysis

In an overall analysis of the strict metrics, we observe that all models present a similar behavior, with a massive performance drop as boundary-expanded mentions are introduced into the training set. However, by examining the curves shaped by the evaluation points, shown in items *a*, *b* and *c* of Figure 4, we can highlight some aspects.

We noted that the models show some robustness against the boundary relaxation noise. While

[2] https://github.com/nicola-decao/efficient-autoregressive-EL

[3] https://github.com/studio-ousia/luke

[4] Implementation available at: https://github.com/gabrielandrade2/VanillaNER

[5] https://huggingface.co

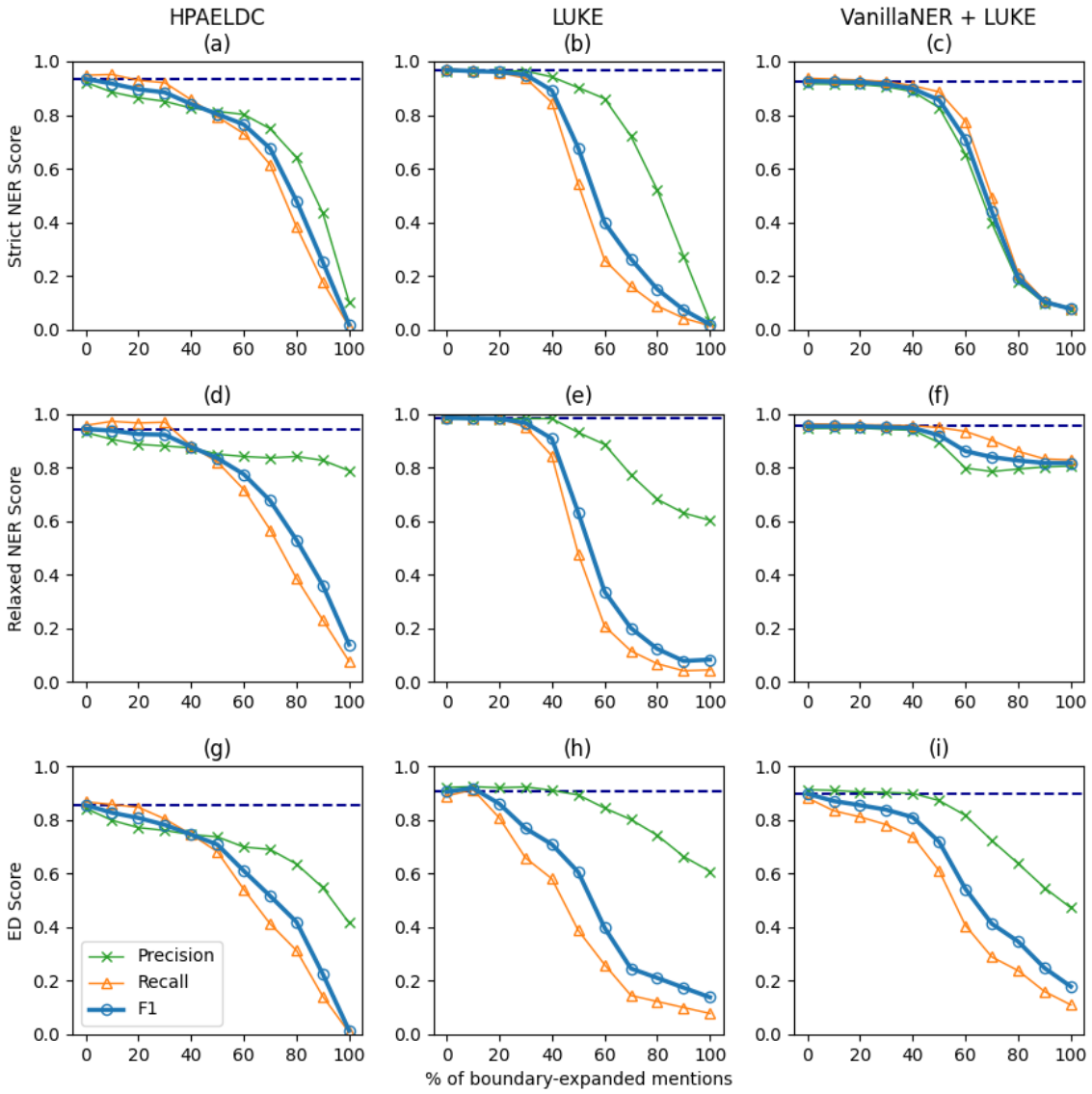

Figure 4: Performance scores for HPAELDC (items *a*, *d* and *g*), LUKE (items *b*, *e* and *h*) and VanillaNER (items *c*, *f* and *i*) across various dataset noise levels. Dashed lines represent the F-score on the Original dataset training.

HPAELDC (Fig. 4.a) appears to be more susceptible to noise, LUKE (Fig. 4.b) and VanillaNER (Fig. 4.c) keep nearly unaltered performance up to the point where 40% of the total mentions in the training set were expanded (less than 8% and 3% of performant e degradation, respectively).

From this point, an accentuated performance decline is perceived in all cases. Yet, the point where the model's score plummets varies between the approaches. For instance, LUKE loses over 42% of its performance in the 50-70% range. Meanwhile, HPAELDC, despite the initial decay, has massive degradation in the 70-100% noise range, with over 70% relative performance drop.

Considering the relaxed metrics presented in Figure 4, items *d* to *f*, we can remark on the distinct behavior of all systems. We observed that both SoTA approaches show a greater performance drop than our baseline model when considering inaccurate boundary extraction. In this scenario, VanillaNER conveys only a small relative degradation, even in noisier training environments (up to 15% lost in F-score when all training mentions were expanded). The relatively elevated precision rates portrayed by the SoTA models denote the need for a higher confidence level before indicating a NE detection.

5.2 NER matching type analysis

To better interpret the performance trend, we further inspected the models' output predictions and

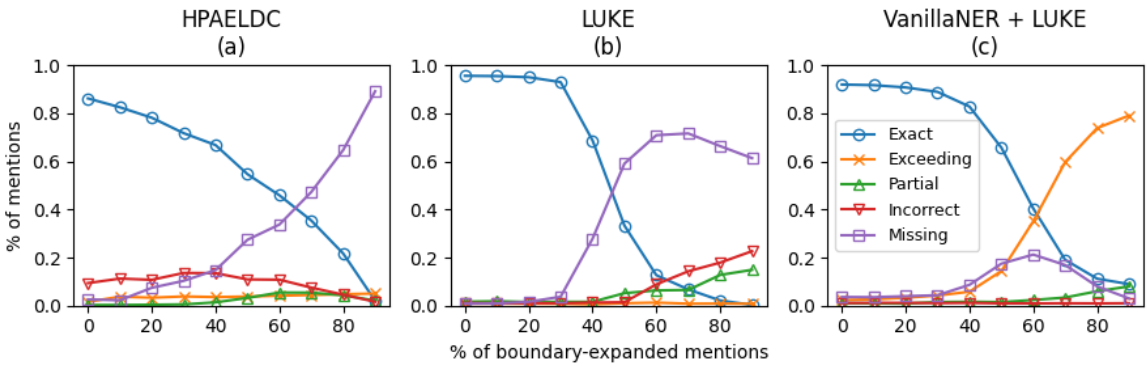

Figure 5: NER matching types for various dataset noise levels. Values are an average from the results of 10 evaluated models and presented in percentages in relation to the total number of gold mentions.

investigated the matching types produced by each model. From the results shown in Figure 5 (complete table is presented in the Appendix A.5), we noticed that both SoTA models tend to miss NE mentions as noisier training dataset instances are applied, in comparison to our approach. This finding also explains the heavier drop in the relaxed scores demonstrated by them. On the other hand, as the noise level is raised, VanillaNER (Fig 5.c) increasingly outputs a higher number of *exceeding matches*, indicating that it successfully adapts to the new domain presented by the boundary-expanded entities and outputs mentions that are closer to what is observed in the training and validation sets.

Also, it is interesting to remark that although the missed-mention rate of this approach increases, it is nowhere near the level presented by the other two models. This distinct behavior is showcased in Table 2, where we present the output processing of an example document.

We attribute the high rate of missing mentions to the greater complexity of SoTA NER approaches. We believe that the complexity of these models, while improving resilience and unlocking higher performance levels in an optimal scenario, hinders their ability to fit into unseen domains. As far as our results show, the SoTA models never actually *learn* the peculiar boundary information encoded in the boundary-expanded annotations, attempting to minimize the noise effect in its language model.

5.3 ED performance analysis

ED performance evaluation, presented in Figure 4, items *g* to *i*, have scores bounded to how well the earlier NER phase performed, as it feeds the disambiguation component. We are aware bad NER output quality may obfuscate the isolated evaluation of ED methods, yet the results give an overview of the pipeline as a whole.

The observed metrics indicate that the disambiguation step is sensitive to imprecise boundaries. In contrast to NER models, ED is not resilient against noise. The increasing ratio of boundary-expanded mentions directly correlates to lower recall scores in all tested approaches. This characteristic denotes that boundary relaxation severely affects the overall capability of recognizing the entity. It can be attributed in part due to missing mention detections but also to the ED model's incapability of generating candidate entities for linking. This can be seen in the example shown in Table 2, where many detected mentions are associated with the *NIL* tag, denoting that not even a potential link to the KB was found.

In further analysis of the output from a few sentences, we verified a similar scenario for both ED approaches. Even though the overall noise level of the training dataset is elevated, whenever the mentions fed by the NER component have the exact boundaries as the mention span both HPAELDC and LUKE were still able to confidently predict the correct KB entity, as shown by the high precision scores.

5.4 ED vocabulary dependency

To better understand the ED component behavior under different inputs, we computed the accuracy score separately for each valid NER matching type (*Exact*, *Exceeding*, and *Partial*). For the sake of simplicity, we averaged the scores across all predictions from all different noise levels.

We remark on the responses both evaluated ED approaches have when dealing with non-exact mention matches, depicted by the results in Table 3.

Original Annotation	**[Japan]** (Japan National Football Team) began the defence of their **[Asian Cup]** (1996 Asian Cup) title with a lucky 2-1 win against **[Syria]** (Syria National Football Team) in a Group C championship match on Friday. **[China]** (People's Republic of China) controlled most of the match [...] until the 78th minute when **[Uzbek]**(Uzbekistan National Football Team) striker **[Igor Shkvyrin]** (Igor Shkvyrin) took advantage [...].
HPAELDC (De Cao et al., 2021)	**[Japan began]** (Japan National Rugby Union Team) the defence of their **[Asian Cup title]** (Coppa Italia) with a lucky 2-1 win against Syria in a **[Group C]** (Middle East) championship match on Friday. **[China]** (People's Republic of China) controlled most of the match [...] until the 78th minute when **[Uzbek]** (Uzbekistan National Football Team) striker Igor Shkvyrin took advantage [...].
LUKE (Yamada et al., 2020)	Japan began the defence of their Asian Cup title with a lucky 2-1 win against **[Syria]** (Syria National Football Team) in a Group C championship match on Friday. **[China]** (People's Republic of China) controlled most of the match [...] until the 78th minute when Uzbek striker Igor Shkvyrin took advantage [...].
VanillaNER + LUKE	**[Japan began]** (NIL) the defence of their **[Asian Cup title]** (NIL) with a lucky 2-1 win **[against Syria in]** (Syria National Football Team) a **[Group C]** (NIL) championship match on Friday. **[China controlled]** (Syria National Football Team) most of the match [...] until the 78th minute when **[Uzbek]** (Uzbekistan National Football Team) **[striker Igor Shkvyrin took]** (Igor Shkvyrin) advantage [...].

Table 2: Comparison of an example document processed by all models trained with a dataset at 70% of noise level. Detected spans are presented in brackets with extra words matched by the NER component marked in blue, whilst undetected and incorrect mentions are highlighted in red. The KB unique identifier is shown in parenthesis and underlined. Incorrectly recognized identifiers are highlighted in orange.

Both models exhibit low effectiveness when attempting to link NER detections with imprecise boundaries, as given by the low *Exceeding* and *Partial match* accuracy values when compared to *Exact matches*.

Upon a closer error analysis of the mistakes for non-exact matches, we noted that the evaluated ED models are strongly reliant on the provided entity vocabulary, especially for the candidate generation step. For most of the *Exceeding matches* predicted, HPAELDC produced a set of candidates that didn't slightly resemble the mention surface form, while LUKE failed to even list potential matches from the KB. On the other hand, *Partial matches* can still be matched correctly, as the lack of one or more words from the mention surface form can still produce associations within the known vocabulary. We believe these effects could be mitigated by an approximate string retrieval method, though prediction time may be affected on such large KBs.

The high values for *Exact match* evaluation indicate that both approaches correctly identify entities even in noisy environments, given that the provided mention spans have precise boundaries. Important to note that when ranking the entity candidates not only the surface form is used, but also the NE context is crucial for accurate prediction. Yet, training in a noisy environment seems to have an insignificant effect on the model's contextual understanding as long as the mention boundaries and, therefore, the NE surface form matches the model's vocabulary.

Model	Matching type		
	Exact	*Exceeding*	*Partial*
HPAELDC	0.827	0.342	0.528
LUKE	0.925	0.015	0.409
VanillaNER + LUKE	0.874	0.048	0.152

Table 3: ED accuracy by NER matching type. Values are an average of the overall produced results (Original dataset + all noise levels.

6 Conclusion

In this study, we investigated the impact of relaxed entity boundary annotation on model performance by conducting a case study with three different EL approaches. We introduced noise into a popular dataset by expanding the boundaries of some annotated entity mentions, trained models on the

produced noisy data, and evaluated their output.

For NER components, we verified that model complexity plays a significant role in how its influenced by noise. While complex models may achieve higher performance, they had reduced flexibility to adapt to the noisy environment, thus being outperformed by our simpler approach in this scenario. We also noted that those models are resilient against the boundary-expanded environment until more than half of the training annotations are "noisy".

We also observed that ED components are sensitive to imprecise entity mentions and strongly reliant on their training vocabulary. We show that both ED approaches evaluated can still correctly predict identifiers even when trained in noisy environments, given that the detected mention boundaries are precise. On the other hand, when boundary imprecision produces out-of-vocabulary mentions, it hinders performance heavily.

6.1 Future Work

Our goal with this case study was to assess the impact suffered by an EL system without the expense of a real annotation process. Considering that we now understand the impact, we intend to continue investigating the possible gains in speed and workload reduction by applying a boundary-relaxed guideline to a corpus annotation process.

We also intend to evaluate the boundary relaxation effect in specific language domains, such as medical texts. Since EL in medical vocabulary is commonly supported by standardized terminologies, typically restricted to a smaller set of words, evaluating the resilience of other linking methodologies (approximate string matching, for instance) to such noise would also be interesting.

Limitations

We acknowledge that the randomness of our noise generation procedure may generate a new entity span that can be considered unnatural (for example, adding prepositions to a city name). Such aspect of our method may have some impact on the performance levels measured, as distinct types of annotation mistakes can affect model performance differently. In this case, added noisy words that are improbable to be part of the entity may be more easily “ignored” by the model, while ambiguous additions can lead to a mistake. Still, this approach can be used as a baseline and a more sophisticated performance assessment using more complex modeling or even real annotation inaccuracy could be done in future work.

In addition, the constraint of expanding the mention boundary by a single token should be also taken into consideration. The reason for this design choice was not only based on narrowing the analysis spectrum by reducing the amount of data we had to investigate but also on time constraints, as the training procedure of multiple models on an even larger number of noisy dataset instances would escalate quickly. However, now conscious of the behavior relaxed annotation has on model behavior, it would be interesting to evaluate how this tendency is transformed by introducing even more unnecessary adjacent context into the annotations.

A last limitation that can be pointed out is that we only evaluated the noise effect in a single dataset. There are other widely adopted benchmarks for EL, such as MSNBC (Cucerzan, 2007) and ClueWeb (Cao et al., 2007), which could be used in this work. However, we feel it would be more interesting to juxtapose with other textual domains, especially those with specific jargon and NEs, such as the medical domain.

Acknowledgements

This work was supported by JST CREST Grant Number JPMJCR22N1 and JSPS KAKENHI Grant Number JP19H01118, Japan.

References

Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. 2007. Learning to rank: From pairwise approach to listwise approach. In *Proceedings of the 24th International Conference on Machine Learning*, ICML ’07, page 129–136, New York, NY, USA. Association for Computing Machinery.

Wendy W Chapman, Prakash M Nadkarni, Lynette Hirschman, Leonard W D’avolio, Guergana K Savova, and Ozlem Uzuner. 2011. Overcoming barriers to nlp for clinical text: The role of shared tasks and the need for additional creative solutions.

Hong Chen, Zhenhua Fan, Hao Lu, Alan Yuille, and Shu Rong. 2018. PreCo: A large-scale dataset in preschool vocabulary for coreference resolution. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 172–181, Brussels, Belgium. Association for Computational Linguistics.

Yejin Choi and Claire Cardie. 2008. Learning with compositional semantics as structural inference for

subsentential sentiment analysis. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 793–801, Honolulu, Hawaii. Association for Computational Linguistics.

Silviu Cucerzan. 2007. Large-scale named entity disambiguation based on Wikipedia data. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 708–716, Prague, Czech Republic. Association for Computational Linguistics.

Hong-Jie Dai, Richard Tzong-Han Tsai, and Wen-Lian Hsu. 2014. Joint learning of entity linking constraints using a markov-logic network. In *International Journal of Computational Linguistics & Chinese Language Processing, Volume 19, Number 1, March 2014*.

Nicola De Cao, Wilker Aziz, and Ivan Titov. 2021. Highly parallel autoregressive entity linking with discriminative correction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7662–7669, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Milan Dojchinovski, Dinesh Reddy, Tomáš Kliegr, Tomáš Vitvar, and Harald Sack. 2016. Crowdsourced corpus with entity salience annotations. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3307–3311.

Omid Ghiasvand and Rohit J. Kate. 2018. Learning for clinical named entity recognition without manual annotations. *Informatics in Medicine Unlocked*, 13:122–127.

Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. 2011. Robust disambiguation of named entities in text. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 782–792, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Nathalie Japkowicz and Mohak Shah. 2011. *Evaluating Learning Algorithms: A Classification Perspective*. Cambridge University Press, USA.

Nikolaos Kolitsas, Octavian-Eugen Ganea, and Thomas Hofmann. 2018. End-to-end neural entity linking. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pages 519–529, Brussels, Belgium. Association for Computational Linguistics.

Robert Leaman, Ritu Khare, and Zhiyong Lu. 2015. Challenges in clinical natural language processing for automated disorder normalization. *Journal of Biomedical Informatics*, 57:28–37.

Jiacheng Li, Haibo Ding, Jingbo Shang, Julian McAuley, and Zhe Feng. 2021. Weakly supervised named entity tagging with learnable logical rules. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4568–4581, Online. Association for Computational Linguistics.

Kun Liu, Yao Fu, Chuanqi Tan, Mosha Chen, Ningyu Zhang, Songfang Huang, and Sheng Gao. 2021. Noisy-labeled NER with confidence estimation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3437–3445, Online. Association for Computational Linguistics.

Mónica Marrero, Julián Urbano, Sonia Sánchez-Cuadrado, Jorge Morato, and Juan Miguel Gómez-Berbís. 2013. Named entity recognition: Fallacies, challenges and opportunities. *Computer Standards & Interfaces*, 35(5):482–489.

Tatyana Shmanina, Ingrid Zukerman, Antonio Jimeno Yepes, Lawrence Cavedon, and Karin Verspoor. 2013. Impact of corpus diversity and complexity on NER performance. In *Proceedings of the Australasian Language Technology Association Workshop 2013 (ALTA 2013)*, pages 91–95, Brisbane, Australia.

Ruixue Tang, Yanping Chen, Yongbin Qin, Ruizhang Huang, Bo Dong, and Qinghua Zheng. 2022. Boundary assembling method for joint entity and relation extraction. *Knowledge-Based Systems*, 250:109129.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Richard Tzong-Han Tsai, Shih-Hung Wu, Wen-Chi Chou, Yu-Chun Lin, Ding He, Jieh Hsiang, Ting-Yi Sung, and Wen-Lian Hsu. 2006. Various criteria in the evaluation of biomedical named entity recognition. *BMC bioinformatics*, 7(1):1–8.

Özlem Uzuner, Brett R South, Shuying Shen, and Scott L DuVall. 2011. 2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association*, 18(5):552–556.

Albert Weichselbraun, Adrian M.P. Brasoveanu, Philipp Kuntschik, and Lyndon J.B. Nixon. 2019. Improving named entity linking corpora quality. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 1328–1337, Varna, Bulgaria. INCOMA Ltd.

Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. 2020. LUKE: Deep contextualized entity representations with entity-aware self-attention. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

Ikuya Yamada, Koki Washio, Hiroyuki Shindo, and Yuji Matsumoto. 2022. Global entity disambiguation with BERT. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3264–3271, Seattle, United States. Association for Computational Linguistics.

Nian Yongming, Chen Yanping, Qin Yongbin, Huang Ruizhang, Tang Ruixue, and Hu Ying. 2022. A joint model for entity boundary detection and entity span recognition. *Journal of King Saud University-Computer and Information Sciences*, 34(10):8362–8369.

Wenzheng Zhang, Wenyue Hua, and Karl Stratos. 2021. EntQA: Entity linking as question answering. *arXiv preprint arXiv:2110.02369*.

Enwei Zhu and Jinpeng Li. 2022. Boundary smoothing for named entity recognition. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7096–7108, Dublin, Ireland. Association for Computational Linguistics.

Yu Zhu, Yingchun Ye, Mengyang Li, Ji Zhang, and Ou Wu. 2022. Investigating annotation noise for named entity recognition. *Neural Computing and Applications*, pages 1–15.

Özge Sevgili, Artem Shelmanov, Mikhail Arkhipov, Alexander Panchenko, and Chris Biemann. 2022. Neural entity linking: A survey of models based on deep learning. *Semantic Web*, 13(3):527–570.

A Appendix

A.1 Amount of entities unified by boundary expansion

Noise Level	Merged entities
10%	0.40%
20%	0.99%
30%	1.81%
40%	2.89%
50%	4.19%
60%	5.74%
70%	7.43%
80%	9.36%
90%	11.67%
100%	14.04%

Table 4: Percentage of entities that were unified by the boundary expansion process for each noise level.

A.2 Example of matching types

Match type	Example
Exact match	[**Yokohama F.C.**] striker [**Kazuyoshi Miura**] scored twice.
Exceeding match	[**Yokohama F.C.**] striker [**Kazuyoshi Miura** scored] twice.
Partial match	[**Yokohama F.C.**] striker [**Kazuyoshi**] **Miura** scored twice.
Invalid match	[**Yokohama F.C.**] striker **Kazuyoshi Miura** [scored] twice.
Missing match	[**Yokohama F.C.**] striker **Kazuyoshi Miura** scored twice.

Table 5: Example of matching types. Correct entity spans are marked in bold, blue and red bracket represent different mention detections

A.3 Training Parameters

	HPAELDC (De Cao et al., 2021)	LUKE (Yamada et al., 2020)	VanillaNER
Max epochs	100	5 (NER) / 2 (ED)	10
Training batch size	32	8 (NER) / 16 (ED)	16
Learning rate	1e-4 (Longformer)/ 1e-3 (other components)	1e-5 (NER) / 2e-5 (ED)	1e-5
Optimizer	Adam	AdamW	AdamW
Model selection	Highest validation micro F-score	Highest validation accuracy	Lowest validation loss
Training time (h)	1.5	3	0.5

Table 6: Hyper-parameters used in the training procedure of the experiment.

A.4 Detailed scoring results

Model	Noise level	Strict			Relaxed		
		Precision	*Recall*	*F-score*	*Precision*	*Recall*	*F-score*
HPAELDC (De Cao et al., 2021)	Original	**0.922**	0.948	**0.935**	**0.930**	0.957	**0.943**
	10%	0.885	**0.951**	0.917	0.905	**0.972**	0.937
	40%	0.826	0.857	0.841	0.873	0.881	0.877
	70%	0.749	0.615	0.675	0.836	0.568	0.676
	100%	0.102	0.010	0.018	0.788	0.076	0.139
LUKE (Yamada et al., 2020)	Original	**0.964**	**0.972**	**0.968**	**0.982**	**0.989**	**0.985**
	10%	0.961	0.966	0.963	0.981	0.986	0.983
	40%	0.942	0.846	0.891	0.982	0.885	0.931
	70%	0.721	0.161	0.263	0.772	0.333	0.466
	100%	0.030	0.014	0.019	0.603	0.279	0.382
VanillaNER	Original	**0.916**	**0.937**	**0.927**	0.945	**0.962**	**0.954**
	10%	0.915	0.934	0.924	**0.946**	0.961	**0.954**
	40%	0.887	0.909	0.898	0.939	0.956	0.947
	70%	0.396	0.494	0.440	0.785	0.901	0.839
	100%	0.075	0.079	0.077	0.805	0.828	0.816

Table 7: Summary of NER evaluation scores for various dataset noise levels. Values are an average of the scores from the evaluation of 10 models. The highest values for each model are marked in bold.

Model	Noise Level	*Precision*	*Recall*	*F-score*
HPAELDC (De Cao et al., 2021)	Original	**0.843**	**0.867**	**0.855**
	10%	0.798	0.858	0.827
	40%	0.745	0.746	0.746
	70%	0.650	0.413	0.505
	100%	0.376	0.007	0.014
LUKE (Yamada et al., 2020)	Original	0.921	0.889	0.905
	10%	**0.924**	**0.913**	**0.918**
	40%	0.910	0.580	0.709
	70%	0.801	0.145	0.246
	100%	0.608	0.078	0.138
VanillaNER + LUKE	Original	**0.913**	**0.882**	**0.897**
	10%	0.910	0.834	0.870
	40%	0.899	0.707	0.792
	70%	0.823	0.389	0.528
	100%	0.772	0.190	0.305

Table 8: Summary of ED evaluation scores for various dataset noise levels. Values are an average of the scores from the evaluation of 10 models. The highest values for each model are marked in bold.

A.5 Breakdown of NER matching types

Model	Noise level	Matching Type (%)					# of mentions
		Exact	*Exceeding*	*Partial*	*Invalid*	*Missing*	
HPAELDC (De Cao et al., 2021)	Original	88.34	0.37	0.31	6.81	4.16	4813
	10%	86.18	1.58	0.26	9.38	2.61	4949
	40%	71.72	3.83	0.52	13.60	10.33	5191
	70%	45.88	4.14	5.46	10.80	33.72	5028
	100%	2.08	5.05	2.13	1.60	89.14	4558
LUKE (Yamada et al., 2020)	Original	96.28	0.97	0.67	1.51	0.58	4993
	10%	95.56	1.13	0.86	1.63	0.81	4941
	40%	92.95	0.99	0.77	1.58	3.70	4934
	70%	8.68	8.86	1.27	2.13	79.07	5360
	100%	0.31	22.66	0.79	14.90	61.34	6316
VanillaNER	Original	92.18	2.05	1.05	1.05	3.68	4937
	10%	91.94	2.32	1.06	1.09	3.60	4939
	40%	88.88	4.93	0.79	1.07	4.33	4938
	70%	40.39	37.09	0.36	0.93	21.23	4932
	100%	8.94	83.75	0.36	1.01	5.94	4935

Table 9: Breakdown of the NER prediction output by matching types.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
In the Limitations section at the end of the paper (Section 7)

☒ A2. Did you discuss any potential risks of your work?
In our paper, we simply evaluate the performance of models to address the task of Entity Linking in simulated scenarios with noisy data. We cannot think of any potential risks of our work.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Abstract and Section 1 (Introduction)

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Section 4 (Dataset and models subitems)

☑ B1. Did you cite the creators of artifacts you used?
Section 4

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Section 4

☒ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
We do not discuss it explicitly in the paper. However, the artifacts are available under Apache License 2.0 and MIT License, which gives permission on Commercial use, Modification, Distribution and Private use.

☒ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
The dataset we used is a widely common open-source benchmark for Entity Linking tasks, so we didn't do any checking steps

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Section 4

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Section 4

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

C ☑ Did you run computational experiments?

Section 5

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
We report each model's total number of parameters and computational infrastructure used in Section 4. We report the computational times in Appendix A.2.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
We discuss the experimental setup in Section 4. We also report the hyperparameters used for training in Appendix A.2.

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Section 5, Appendix A.3, A.4, and A.5. In Section 3 we inform that we use averaged values. We do not report the standard variation as the obtained values are very small and irrelevant.

☒ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
No such packages were used

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.