A Study on Entity Linking Across Domains: Which Data is Best for Fine-Tuning?

Hassan Soliman[1,2]   **Heike Adel**[1]

Mohamed Gad-Elrab[1]   **Dragan Milchevski**[1]   **Jannik Strötgen**[1]

[1]Bosch Center for Artificial Intelligence, Renningen, Germany
[2]Saarland University, Saarbrücken, Germany
[1]`firstname.lastname@de.bosch.com`
[2]`s8hasoli@stud.uni-saarland.de`

Abstract

Entity linking disambiguates mentions by mapping them to entities in a knowledge graph (KG). One important question in today's research is how to extend neural entity linking systems to new domains. In this paper, we aim at a system that enables linking mentions to entities from a general-domain KG and a domain-specific KG at the same time. In particular, we represent the entities of different KGs in a joint vector space and address the questions of which data is best suited for creating and fine-tuning that space, and whether fine-tuning harms performance on the general domain. We find that a combination of data from both the general and the special domain is most helpful. The first is especially necessary for avoiding performance loss on the general domain. While additional supervision on entities that appear in both KGs performs best in an intrinsic evaluation of the vector space, it has less impact on the downstream task of entity linking.

1 Introduction

Entity linking, i.e., the task of disambiguating mentions in text by linking them to entities of a knowledge graph (KG), is key to many semantic applications, such as KG population, question answering or information retrieval (Sevgili et al., 2021). In the context of KGs, a domain is characterized, i.a., by the set and distribution of entities (Onoe and Durrett, 2020). For KGs from special domains, the availability of annotated data for training entity linking is limited. Thus, there is a need for methods that work across domains in low-resource settings, such as transfer or few-shot learning techniques (Hedderich et al., 2021).

Given a KG from a special domain, it is useful for many applications to not treat this KG in isolation but still be able to link mentions to the general domain as well. Figure 1 illustrates this. Without combining the KGs *Wikipedia* and *Doctor Who*, it would not be possible to link all mentions of the
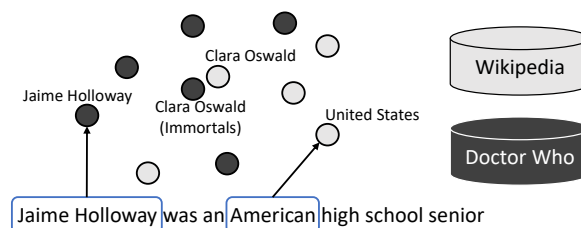

Figure 1: Illustration of entity linking to two KGs (Wikipedia and Doctor Who) at the same time by representing the entities of both graphs in a joint vector space. Colors indicate the KG to which the entities belong.

example sentence to their respective entities. Early works prior to neural entity linking (Hoffart et al., 2011) allow linking to multiple KGs by combining the KGs before applying the methods. In the context of neural networks, a more elegant way is to combine different KGs via a *joint vector space* (Gupta et al., 2017). This also enables us to learn similar embeddings for overlapping entities, i.e., entities that appear in more than one KG, which would arguably be more difficult when only uniting triple sets. In Figure 1, the entities "Clara Oswald" and "Clara Oswald (Immortals)" are an example of overlapping entities.

In this paper, we aim at methods for adding a KG from a specific target domain into an existing vector space from a source-domain KG and fine-tuning the joint vector space to improve the entity linking results. While some recent work has considered zero-shot entity linking (Logeswaran et al., 2019; Wu et al., 2020), a systematic investigation on which data sources are most useful for fine-tuning entity linking systems, is still missing. Thus, the first research question we address is: *Which data is best suited for fine-tuning joint vector spaces of KGs?*

Furthermore, it is unclear how fine-tuning on the target domain affects the vector space of the source-domain KG. Thus, the second research question we pose is: *Does fine-tuning harm performance of entity linking on the general domain?*

To answer these research questions, we present a systematic investigation of the impact of different information sources on the vector space (intrinsic evaluation) as well as on the entity linking performance (extrinsic evaluation). Further, we will publish the list of overlapping entities that we created along with this paper to ensure reproducibility.

2 Related Work

As in other fields of natural language processing, deep learning became the predominant approach in entity linking (He et al., 2013; Sun et al., 2015; Francis-Landau et al., 2016; Yamada et al., 2016; Gupta et al., 2017; Kolitsas et al., 2018). In today's research, the usage of pre-trained language models, such as BERT (Devlin et al., 2019), is particularly popular (Peters et al., 2019; Logeswaran et al., 2019; Humeau et al., 2020; Wu et al., 2020).

In this paper, we aim at a neural entity linking system which allows linking mentions to more than one KG by creating a joint vector space for entity representations from different KGs. Related to this, Gupta et al. (2017) propose a method to create a joint vector space for entities from different sources that are represented by different means, such as descriptions, contexts or fine-grained types. In the context of entity linking across domains, Onoe and Durrett (2020) build a domain-independent system that relies on fine-grained entity types. In contrast, Logeswaran et al. (2019) and Wu et al. (2020) utilize descriptions of entities from KGs to obtain entity representations. In particular, Logeswaran et al. (2019) propose domain-adaptive pre-training to apply entity linking to unseen entities from a KG of a new domain. Wu et al. (2020) build on that work but train their model only on labeled data from a general domain (Wikipedia). Vyas and Ballesteros (2021) generalize those models and allow them to handle arbitrary KGs with entities represented by an arbitrary set of attribute-value pairs.

In contrast to those works, we also take into account overlapping entities between the two KGs and study which impact fine-tuning on different data sources has on the joint vector space as well as on entity linking performance.

3 Linking Model and Extension Method

In this section, we detail the entity linking model and describe how the model can be extended to a new domain.

3.1 Entity Linking Model

To be able to directly compare with state-of-the-art related work, we build upon the entity linking model proposed by Wu et al. (2020). It consists of three parts (context encoder, candidate encoder and cross-encoder) which are used in two phases (candidate generation and candidate ranking). Note that our extension approach is independent of the underlying system though.

Candidate generation. In this step, the context encoder creates a vector representation for a mention given a textual context. Similarly, the candidate encoder embeds a candidate entity from the knowledge graph given its textual description. For both model parts, a BERT encoder is used and the CLS token serves as the output embedding. For candidate generation, the k most similar entities to a given mention are retrieved where similarity is measured by cosine similarity between the entity and the mention embeddings.

Candidate ranking. For candidate ranking, the cross-encoder estimates how likely a mention represents a candidate entity. For this, a third BERT model is used that receives as input the concatenation of the textual context of the mention and the title and description of the candidate entity. Its CLS token is then fed into a feed-forward layer to compute a score that is trained to be higher for the correct candidate entity than for wrong candidate entities.

3.2 Extension to New Domains

To extend the model to a new domain, we fine-tune the weights θ of the context and candidate encoders.

Information Sources for Fine-Tuning. In our experiments, we investigate which data or which set of data is most promising for fine-tuning. For this, we use the following information sources: (i) data annotated with entities from the KG of the *target domain* (T), (ii) additional data that is annotated with entities from the KG of the *source domain* (S), and (iii) a list of *overlapping entities* between the KG from the source domain and the KG of the target domain (O). The following paragraphs describe how the data sources are used for fine-tuning.

Fine-Tuning Loss Functions. For fine-tuning on data annotated with entity information from a KG

	Domain	Entities	Mentions			Overlapping (O)	
			Train*	Dev*	Test	Candidates	Filtered
T	American Football	31,929	3,000	320	578	24,074	22,928
	Doctor Who	40,281	6,360	640	1,334	10,458	3,611
	Fallout	16,992	2,500	320	466	2,876	752
	Final Fantasy	14,044	4,360	640	1,041	1,495	413
S	Wikipedia (Reddit)	5,903,538	7,711	409	1,328	-	-

Table 1: Statistics for the datasets used in our experiment (* corresponds to the fine-tuning phase).

(i.e., settings S and T), we use the following loss function:

$$s(a, b) := a^\top b$$

$$L_\theta = \sum_{m,e \in D} \big(-s(v_m, v_e) + \log \sum_{c \in C_e} \exp(s(v_m, v_c)) \big) \tag{1}$$

where D is a dataset, annotated with mentions m and their corresponding entities e, v_m is the representation of the context encoder of mention m in a textual context, v_e is the representation of the candidate encoder of the textual description of entity e from the KG and C_e is a randomly sampled batch of negative entities from the KG.

For fine-tuning on the list of overlapping entities (i.e., setting O), we use the following loss function:

$$L_\theta = \sum_{e_{\text{KG1}}, e_{\text{KG2}} \in D} \big(-2 \cdot s(v_{e_{\text{KG1}}}, v_{e_{\text{KG2}}})$$
$$+ \log \sum_{c \in C_{e_{\text{KG2}}}} \exp(s(v_{e_{\text{KG1}}}, v_c))$$
$$+ \log \sum_{c \in C_{e_{\text{KG1}}}} \exp(s(v_{e_{\text{KG2}}}, v_c)) \big) \tag{2}$$

where D is the list of overlapping entities, e_{KG1} is an entity that appears in KG 1, e_{KG2} is its counter entity from KG 2 and $v_{e_{\text{KG1}}}$ and $v_{e_{\text{KG2}}}$ are the representations of their textual descriptions from the two KGs, and s is defined as in Equation 1. $C_{e_{\text{KG2}}}$ and $C_{e_{\text{KG1}}}$ are randomly sampled batches of negative entities from the list of overlapping entities (that are from KG 2 and KG 1, respectively, and do not overlap with $v_{e_{\text{KG1}}}$ and $v_{e_{\text{KG2}}}$, respectively). This loss function encourages overlapping entities from KG 1 and KG 2 to have similar vector representations in the joint vector space while it pushes representations of other entity pairs further apart.

Combination of Information Sources. We also experiment with combinations of the different information sources described above. In particular when combining settings S and T, in each epoch, we first present batches from S to the model followed by

batches from T. When adding O to a combination, we first fine-tune the model on S and/or T (using loss function 1) and then continue fine-tuning on O (using loss function 2).

4 Experiments and Results

In this section, we present our study and report the effects of fine-tuning on the latent representation as well as on the downstream task of entity linking.

Baseline. We use the neural entity linking model BLINK (Wu et al., 2020) as our baseline model.[1]

Fine-tuning Configurations. We experiment with the following combinations of the information sources presented in Section 3.2: S, T, TO, TS, TOS.[2] Hyperparameters are provided in Section A of the appendix.

4.1 Data

Data from target domain (T). The experimental setup requires *domain-specific* entity linking data which is split into fine-tuning and test set. To the best of our knowledge, there is no benchmark available for this. Therefore, we adopt the Wikia dataset for zero-shot entity linking across domains (Logeswaran et al., 2019). We select four domains (*American Football*, *Doctor Who*, *Fallout*, *Final Fantasy*) and randomly split each domain into fine-tuning (train and dev) and test sets (see top part of Table 1). Throughout the experiments, we consider Wikipedia as the *source-domain KG* and one of the domains from Wikia as the *target-domain KG*.

Data from source domain (S). As additional contextual data for source-domain entities, we adopt the Reddit dataset (Botzer et al., 2021) that contains Reddit blog posts with mentions linked to

---

[1]It has been trained on an English Wikipedia dump from May 2019, using over 5.9M pages as entities (page titles are used as entity names and summary paragraphs as descriptions) and around 9M Wikipedia interlinks as mention-entity annotations. For further details, please see Wu et al. (2020).

[2]We focus on combinations with T since combinations without T did not perform well in preliminary experiments.

Target KG →	American Football		Doctor Who		Fallout		Final Fantasy	
Model	MRR	ACS	MRR	ACS	MRR	ACS	MRR	ACS
BLINK	0.4991	0.9938	0.4607	0.9650	0.4071	0.9603	0.3623	0.9532
T	0.4982	0.9892	0.3926	0.9095	0.3533	0.9317	0.4136*	0.9515
TO	0.4990	0.9919	**0.4932***	0.9784*	**0.4558***	0.9680*	**0.4400***	0.9628
TS	**0.4999**	**0.9958***	0.4323	0.9605	0.4223*	0.9676*	0.4072*	0.9746*
TOS	0.4995	0.9896	0.4619	**0.9830***	0.4534*	**0.9820***	0.4209*	**0.9791***

Table 2: Intrinsic evaluation of overlapping entities between each domain-specific KG and Wikipedia KG. * shows statistically different results in comparison to BLINK (randomization test, $\alpha = 0.005$ with Bonferroni correction).

	Target KG →	American Football		Doctor Who		Fallout		Final Fantasy	
	Model	AP@1	MAP@10	AP@1	MAP@10	AP@1	MAP@10	AP@1	MAP@10
Eval on target KG	BLINK	0.1747	0.4104	0.4108	0.4810	0.3412	0.4444	0.3833	0.5179
	S	0.1713	0.3732	0.5337*	0.6191*	0.4249*	0.5295*	0.3881	0.5433
	T	**0.2093***	**0.4606***	**0.6169***	0.6925*	0.4313*	0.5510*	0.3871	0.5405
	TO	0.1938	0.4103	0.5697*	0.6558*	0.4485*	0.5590*	0.3439	0.4881
	TS	0.2076	0.4583*	0.6124*	**0.7124***	**0.4657***	**0.5915***	**0.4121**	**0.5710***
	TOS	0.1540	0.3292	0.5345*	0.6149*	0.4227*	0.5405*	0.3910	0.5486*
Eval on source KG	BLINK	0.8479	0.8973	0.8509	0.8985	0.8509	0.8987	0.8494	0.8987
	S	**0.8727**	**0.9051**	0.8750*	**0.9063**	**0.8788***	**0.9089**	0.8758*	0.9070
	T	0.8539	0.8994	0.8209	0.8556	0.8057	0.8464	0.8599	0.8991
	TO	0.8524	0.8953	0.8532	0.8865	0.8381	0.8714	0.8630	0.8956
	TS	0.8607	0.8957	0.8582	0.8976	0.8599	0.8965	0.8788*	**0.9085**
	TOS	0.8170	0.8386	**0.8773***	0.9062	0.8637	0.8858	**0.8795***	0.9038

Table 3: Extrinsic evaluation: entity linking. Source KG is Wikipedia in all cases. For the evaluation on the target KG, the domain-specific test set is used. For the evaluation on the source KG, the Reddit test set is used. * shows statistically different results in comparison to BLINK (randomization test, $\alpha = 0.005$ with Bonferroni correction).

Wikipedia entities. We choose the mentions with gold annotations as test set and the mentions with bronze and silver annotations as fine-tuning set.[3] The bottom part of Table 1 shows statistics on the data from the source domain.

Overlapping entities (O). To obtain overlapping entities between the source KG and each of the domain-specific KGs, we first create a candidate list with strict string matching of the entity name aliases (titles of Wikipedia/Wikia pages) from the source and target KGs. Second, we filter this list based on the semantic similarity of the textual descriptions of the entities. In particular, we embed the descriptions with the Roberta-large sentence transformer model by Reimers and Gurevych (2019) and filter the list of candidate entity pairs based on the cosine similarity between their vectors. We set the matching threshold for the cosine similarity to 0.5. Statistics of the overlapping entities are shown on the right side of Table 1.

To ensure the quality of the extracted lists of overlapping entities, we sample 100 entity pairs per domain and manually check their correctness.

Table 4 shows results. Especially for the domains with the largest number of overlapping entities (*American Football* and *Doctor Who*), the number of correct entity pairs is quite high, indicating the usefulness of that information source for our experiments.

Target KG	Correct	Wrong	Unclear
American Football	100	0	0
Doctor Who	84	10	6
Fallout	79	15	6
Final Fantasy	65	17	18

Table 4: Manual analysis of 100 randomly sampled overlapping entities per target domain. The table shows the number of correctly and wrongly identified overlapping entity pairs. The column Unclear shows the number of pairs for which we did not have enough information to assess their correctness.

4.2 Intrinsic Evaluation of Vector Space

We first investigate the effect of fine-tuning on the latent representations of the entities. Intuitively, the better the fine-tuning, the closer the overlapping entities should be in the space. To assess that, we compute the cosine similarity between the vector of the target-domain entity and the vector of its

counter entity from the source KG for each pair in the list of overlapping entities and aggregate the results to the *Average Cosine Similarity (ACS)*. Thus, ACS reflects the average of cosine similarities between overlapping entities.

In addition, we assess the rank of the counter entity in the list of nearest neighbors in the vector space for each entity of the list of overlapping entities. Ideally, the counter entity should have a high rank. We evaluate this with the *Mean Reciprocal Rank (MRR)*, a measure from information retrieval.

Table 2 compares the embeddings generated by our models to the embeddings generated by the baseline model (BLINK) in the four domains. Most fine-tuning configurations outperform the baseline model. This shows the value of fine-tuning for the joint vector space in general. For all domains except for *American Football*, MRR and ACS are enhanced up to 7.77% and 2.59%, respectively, with fine-tuning on overlapping entities (O) providing most performance gains. For *American Football*, the results are closer to the baseline and fine-tuning on overlapping entities (O) does not enhance performance compared to fine-tuning on source and target data (TS). This could be explained by the larger number of overlapping entities in this domain (see Table 1). In general, the results show that fine-tuning on target data only (T) is not sufficient and especially fine-tuning on overlapping entities (O) helps improving the vector space.[4]

4.3 Entity Linking Results

To evaluate entity linking, we use the standard measures of *average precision for the top-1 entity (AP@1)* and the *mean average precision for the top-10* entities *(MAP@10)*.

In the upper part of Table 3, we report the results of applying the fine-tuned models to a set of unseen documents with entities from the different target domains. As shown in the table, fine-tuning outperforms BLINK. Interestingly, even fine-tuning on source entities (S) helps in three out of four domains when evaluating on the target KG. Training on both target and source KG entities (TS) achieves the best performance for all domains with an increase of up to 20% in both MAP and AP measures.

---

[4]Note that the dataset used for fine-tuning on O is the same that we use for the intrinsic evaluation. The reason is that we only aim at analyzing the effects of fine-tuning on the overlapping entities in the joint vector space without the necessity for generalization to an unseen dataset. The effects on the unseen test sets for entity linking are described in Section 4.3.

Including overlapping entities does not further enhance the performance. An explanation for this could be that the entity linking system does not rely on the candidate generation step alone (which requires a good joint vector space) but in addition uses a cross-encoder in the candidate ranking step that re-evaluates each pair of mention and candidate entity, taking the combination of their contexts into account.

In order to ensure that the fine-tuned model still performs well on mentions of entities from the source KG, we also evaluate it on the test data from Reddit. The results can be found in the bottom part of Table 3. Fine-tuning on the source KG only (S) improves the baseline system BLINK as expected. In contrast, fine-tuning on the target KG only (T) harms the performance on the source KG test set a bit. When the model is trained on a combination of entities from both target and source KGs (TS/TOS), performance on the source-KG test set is enhanced in most cases.

With respect to our research questions, we can conclude that *a combination of data with mentions linked to entities from both the source and the target KG* is most suited for fine-tuning and that *especially adding training data from the source-domain KG* avoids performance loss on the source domain. With this fine-tuning setup, we obtain a robust system that can link mentions to both source and target-domain KGs at the same time.

5 Conclusion

In this paper, we presented a systematic investigation of extending an entity linking system to a new domain by creating a joint vector space. Our results showed that it is helpful to add data from both the source domain and the target domain. While an additional supervision on entities that appear in both knowledge graphs improves the quality of the vector space, it has less impact on the downstream task of entity linking. Additional data linked to the source-domain KG avoids performance loss on the general domain and is, thus, especially useful to achieve a system that can link mentions to both source and target-domain KGs at the same time.

Acknowledgments

References

Nicholas Botzer, Yifan Ding, and Tim Weninger. 2021. Reddit entity linking dataset. *Information Processing & Management*, 58(3).

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.

Matthew Francis-Landau, Greg Durrett, and Dan Klein. 2016. Capturing semantic similarity for entity linking with convolutional neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1256–1261, San Diego, California. Association for Computational Linguistics.

Nitish Gupta, Sameer Singh, and Dan Roth. 2017. Entity linking via joint encoding of types, descriptions, and context. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 2681–2690. Association for Computational Linguistics.

Zhengyan He, Shujie Liu, Mu Li, Ming Zhou, Longkai Zhang, and Houfeng Wang. 2013. Learning entity representation for entity disambiguation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 30–34, Sofia, Bulgaria. Association for Computational Linguistics.

Michael A. Hedderich, Lukas Lange, Heike Adel, Jannik Strötgen, and Dietrich Klakow. 2021. A survey on recent approaches for natural language processing in low-resource scenarios. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2545–2568, Online. Association for Computational Linguistics.

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.

Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. 2020. Polyencoders: Architectures and pre-training strategies for fast and accurate multisentence scoring. In *8th International Conference on Learning Representations, ICLR 2020*, Addis Ababa, Ethiopia.

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.

Lajanugen Logeswaran, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, Jacob Devlin, and Honglak Lee. 2019. Zero-shot entity linking by reading entity descriptions. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3449–3460, Florence, Italy. Association for Computational Linguistics.

Yasumasa Onoe and Greg Durrett. 2020. Fine-grained entity typing for domain independent entity linking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 8576–8583, New York, NY, USA. AAAI Press.

Matthew E. Peters, Mark Neumann, Robert Logan, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. 2019. Knowledge enhanced contextual word representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 43–54, Hong Kong, China. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Özge Sevgili, Artem Shelmanov, Mikhail Arkhipov, Alexander Panchenko, and Chris Biemann. 2021. Neural entity linking: A survey of models based on deep learning. *arXiv:2006.00575*.

Yaming Sun, Lei Lin, Duyu Tang, Nan Yang, Zhenzhou Ji, and Xiaolong Wang. 2015. Modeling mention, context and entity with neural networks for entity disambiguation. In *Proceedings of the 24th International Conference on Artificial Intelligence, IJCAI 2015)*, page 1333–1339, Buenos Aires, Argentina. AAAI Press.

Yogarshi Vyas and Miguel Ballesteros. 2021. Linking entities to unseen knowledge bases with arbitrary schemas. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 834–844. Association for Computational Linguistics.

Ledell Wu, Fabio Petroni, Martin Josifoski, Sebastian Riedel, and Luke Zettlemoyer. 2020. Scalable zero-shot entity linking with dense entity retrieval. In

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 6397–6407, Online. Association for Computational Linguistics.

Ikuya Yamada, Hiroyuki Shindo, Hideaki Takeda, and Yoshiyasu Takefuji. 2016. Joint learning of the embedding of words and entities for named entity disambiguation. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 250–259, Berlin, Germany. Association for Computational Linguistics.

A   Hyperparameters

Hyperparameters	Values
Fine-tuning Batch Size	16
Learning Rate	3e-5
Number of Epochs	5
Candidate Generation: Top K	10
Mention with Context: Max Context Length	128
Entity with Description: Max Description Length	128

Table 5: Hyperparameters for fine-tuning.

All settings, i.e., all combinations of information sources (S, T, TS, TO and TOS) share the same hyperparameters for fine-tuning. They are provided in Table 5.

For learning rate and number of epochs, we followed the proposed values by Wu et al. (2020). In addition, we applied early stopping to store the best model checkpoint based on the model performance on the validation set.

We set the number of candidate entities k to 10 and the maximum number of tokens in the context of mention and entity to 128 in order to save computation costs. Note that we applied the same k and context lengths when evaluating the baseline BLINK model.

Fine-tuning and evaluation was performed on a Tesla V100 GPU. All our experiments were run on a carbon-neutral GPU cluster.[5]

B   Ethical Considerations

We acknowledge the ACL Code of Ethics. In particular, we only use well-known benchmark datasets for our evaluation. Both the Wikia and the Reddit dataset do not include personal data, such as information about the authors of the posts (Logeswaran et al., 2019; Botzer et al., 2021). The list of overlapping entities that we will publish does not contain any privacy-related or IP-related content either.

---

[5]The Bosch Group is carbon neutral. Administration, manufacturing and research activities do no longer leave a carbon footprint. This also includes GPU clusters on which the experiments have been performed.