Entity Disambiguation with Entity Definitions

Luigi Procopio[1,2*] **Simone Conia**[1] **Edoardo Barba**[1] **Roberto Navigli**[1]
[1] Sapienza NLP Group, Sapienza University of Rome
[2] Sunglasses.ai
luigi.procopio@sunglasses.ai
conia@di.uniroma1.it
{barba,navigli}@diag.uniroma1.it

Abstract

Local models have recently attained astounding performances in Entity Disambiguation (ED), with generative and extractive formulations being the most promising research directions. However, previous works have so far limited their studies to using, as the textual representation of each candidate, only its Wikipedia title. Although certainly effective, this strategy presents a few critical issues, especially when titles are not sufficiently informative or distinguishable from one another. In this paper, we address this limitation and investigate the extent to which more expressive textual representations can mitigate it. We evaluate our approach thoroughly against standard benchmarks in ED and find extractive formulations to be particularly well-suited to such representations. We report a new state of the art on 2 out of the 6 benchmarks we consider and strongly improve the generalization capability over unseen patterns. We release our code, data and model checkpoints at https://github.com/SapienzaNLP/extend.

1 Introduction

Being able to pair a mention in a given text with its correct entity out of a set of candidates is a crucial problem in Natural Language Processing (NLP), referred to as Entity Disambiguation (Bunescu and Paşca, 2006, ED). Indeed, since ED enables the identification of the actors involved in human language, it is often considered a necessary building block for a wide range of downstream applications, including Information Extraction (Ji and Grishman, 2011; Guo et al., 2013; Fatahi Bayat et al., 2022), Question Answering (Yin et al., 2016) and Semantic Parsing (Bevilacqua et al., 2021; Procopio et al., 2021). ED generally occurs as the last step in an Entity Linking pipeline (Broscheit, 2019), preceded by Mention Detection and Candidate Generation, and its approaches have traditionally been divided into two groups, depending on whether co-occurring mentions are disambiguated independently (*local methods*; Shahbazi et al. (2019); Wu et al. (2020); Tedeschi et al. (2021)), or not (*global methods*; Hoffart et al. (2011); Moro et al. (2014); Yamada et al. (2016); Yang et al. (2018)).

Despite the limiting operational hypothesis of independence between co-occurring mentions, local methods have nowadays achieved performances that are either on par or above those attained by their global counterparts, mainly thanks to the advent of large pre-trained language models. In particular, among these methods, generative (De Cao et al., 2021) and extractive (Barba et al., 2022) formulations are arguably the most promising directions, having resulted in large performance improvements across multiple benchmarks. Regardless of their modeling differences, the key idea behind these methods is to part away from the previous classification-based approaches and, instead, adopt formulations that better leverage the original pre-training of the underlying language models. On the one hand, generative formulations tackle ED as a text generation problem and train neural architectures to generate auto-regressively, given a mention and its context, a textual representation of the correct entity. On the other hand, extractive approaches frame ED as extractive question answering: they first concatenate a textual representation of each entity candidate to the original input and then train a model to extract the span corresponding to the correct entity.

Although they have admittedly attained great improvements, both in- and out-of-domain, to the best of our knowledge, previous works on both these formulations have limited their studies to a single type of textual representation for entities, that is, their title in Wikipedia. This strategy, however, presents a number of issues (Barba et al., 2022) and, in particular, often results in representations that

* Work carried out while at the Sapienza University of Rome.

Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pages 1297–1303

are either insufficiently informative, or are even virtually indistinguishable from one another. In contrast to this trend, we address this limitation and explore the effect of more expressive textual representations on state-of-the-art local methods. To this end, we propose complementing Wikipedia titles with their description in Wikidata, such that, for instance, the candidates for *Ronaldo* in *Ronaldo scored two goals for Portugal* would be *Cristiano Ronaldo: Portuguese association football player* and *Ronaldo: Brazilian association football player*, rather than the less informative *Cristiano Ronaldo* and *Ronaldo*, respectively. We test our novel representations on generative and extractive formulations, and evaluate against standard benchmarks in ED, both in and out of domain, reporting statistically significant improvements for the latter group.

2 Method

We now formally introduce ED and the textual representation strategy we put forward. Then, we describe the two formulations with which we implement and test our proposal.

ED with Entity Definitions. Given a mention m occurring in a context c_m, Entity Disambiguation is formally defined as the task of identifying, out of a set of candidates $e_1, \ldots, e_n$, the correct entity e^* that m refers to. Each candidate e typically corresponds to a page in Wikipedia, and, in generative and extractive formulations, is additionally associated with a textual representation $\hat{e}$ describing its meaning. Whereas previous works have considered the title that e has in Wikipedia as $\hat{e}$ (e.g. *Cristiano Ronaldo*), here we focus on more expressive alternatives and leverage Wikidata to achieve this objective. Specifically, we first retrieve the Wikidata description of e (e.g. *Portuguese association football player*). Then, we define as the new representation of e the colon-separated concatenation of its Wikipedia title and its Wikidata description (e.g., *Cristiano Ronaldo: Portuguese association football player*).

Generative Modeling. In our first formulation, we follow De Cao et al. (2021) and frame ED as a text generation problem. Starting from a mention m and its context c_m, we first wrap the span corresponding to m in c_m between two special symbols, namely <*s*> and </*s*>; we denote this modified sequence by $\tilde{c}_m$. Then, we train a sequence-to-sequence model to generate the textual representation $\hat{e}^*$ of the correct entity e^* by learning the following probability:

$$p(\hat{e}^*|\tilde{c}_m) = \prod_{j=1}^{|\hat{e}^*|} p(\hat{e}^*_{(j)}|\hat{e}^*_{(0:j-1)}, \tilde{c}_m)$$

where $\hat{e}^*_{(j)}$ denotes the j-th token of $\hat{e}^*$ and $\hat{e}^*_{(0)}$ is a special start symbol. The purpose of <*s*> and </*s*> is to signal to the model that m is the mention we are interested in disambiguating. As in the reference work, we use BART (Lewis et al., 2020) as our sequence-to-sequence architecture for our experiments and, most importantly, adopt constrained decoding on the candidate set at inference time. Indeed, applying standard decoding methods such as beam search might result in outputs that do not match any of the original candidates; thus, to obtain only valid sequences, at each generation step we constrain the set of tokens that can be generated according to a prefix tree (Cormen et al., 2009) built over the candidate set.

Extractive Modeling. Additionally, we also consider the formulation recently presented by Barba et al. (2022) that frames ED as extractive question answering. Here, $\tilde{c}_m$, defined in the same way as it was for Generative Modeling, above, represents the query, whereas the context is built by concatenating the textual representations $\hat{e}_1, \ldots, \hat{e}_n$ of the candidates $e_1, \ldots, e_n$. A model is then trained to extract the text span that corresponds to e^*. Following the same efficiency reasoning of the authors, we use as our underlying model the Longformer (Beltagy et al., 2020), whose linear attention better scales to this type of long-input formulations. Compared to the above generative method, the benefits of this approach lie in i) dropping the need for a potentially slow auto-regressive decoding process, and ii) enabling full joint contextualization both between context and candidates, and across candidates themselves.

3 Experiments and Results

In order to assess the impact of our proposal in ED, we evaluate how the performances of generative and extractive formulations change when moving from Wikipedia titles to our alternative. To this end, in this Section, we first describe our experimental setting, discussing the datasets, evaluation strategy and comparison systems we adopt. Then, we describe the architecture we use for the two formulations. Finally, we present our findings.

	Dataset	Instances	Candidates	Failures
AIDA	Train	18,448	905,916 / 79,561	5038 / 682
	Validation	4791	236,193 / 43,339	1360 / 296
	Test	4485	231,595 / 46,660	1395 / 323
OOD	MSNBC	656	17,895 / 8336	149 / 72
	AQUAINT	727	23,917 / 16,948	142 / 121
	ACE2004	257	12,292 / 8045	66 / 50
	CWEB	11,154	462,423 / 119,781	3642 / 1265
	WIKI	6821	222,870 / 105,440	1216 / 719

Table 1: Number of instances, candidates and failures to map a Wikipedia title to its Wikidata definition in the AIDA-CoNLL (top) and out-of-domain (OOD, bottom) datasets. For candidates and failures, we report both their total (base) and unique (exponent) number.

3.1 Experimental Setup

Data. We follow the same experimental setting described by De Cao et al. (2021) and use the standard AIDA-CoNLL splits (Hoffart et al., 2011) for training, model selection and in-domain evaluation (AIDA); similarly, we leverage their cleaned version of MSNBC, AQUAINT, ACE2004, WNED-CWEB (CWEB) and WNED-WIKI (WIKI) (Guo and Barbosa, 2018; Evgeniy et al., 2013) for out-of-domain evaluation and use their same candidate sets, which were originally presented by Le and Titov (2018).[1] We match each entity candidate with its item in Wikidata[2] to retrieve the corresponding description. Due to inconsistencies in the datasets and different dump versions, this mapping is not always possible, and, in these cases, we fall back to employing their Wikipedia title alone. We report in Table 1 the number of instances, candidates and mapping failures in each dataset under consideration.

Evaluation. Following previous literature in ED, we compute scores over the test sets in terms of *inKB Micro* F_1. Furthermore, for each system we consider, we calculate the macro average of its performances both over all the test sets (Avg) and over the five out-of-domain datasets only (Avg_{OOD}).

Comparison Systems. We consider the original models presented by De Cao et al. (2021, GENRE) and Barba et al. (2022, ExtEnD), trained on AIDA-CoNLL with Wikipedia titles, as our main natural comparison systems; in particular, for ExtEnD, we evaluate against both its Longformer base ($\text{ExtEnD}_{\text{base}}$) and large ($\text{ExtEnD}_{\text{large}}$) alternatives. Furthermore, to better contextualize the performances we attain within the current landscape of ED, we also include three state-of-the-art systems, namely, the global model of Yang et al. (2018), and the variants of De Cao et al. (2021) and Barba et al. (2022), both pre-trained on BLINK (Wu et al., 2020) before fine-tuning on AIDA-CoNLL. However, we note that, differently from our work, these three systems use additional training data (9M samples) from Wikipedia, whereas, due to computational constraints, we limit our experiments to the sole usage of AIDA-CoNLL ($<$ 20K samples).

3.2 Architectures

For both our formulations, we closely follow the corresponding reference architectures. For the generative method, we use BART (406M parameters) as our underlying sequence-to-sequence model and fine-tune it on AIDA-CoNLL using a 10,000 token batch size, Adam (Kingma and Ba, 2015) as our optimizer and 10^{-5} learning rate, with 500 warm-up steps and linear decay. For the extractive method, we test and evaluate our approach on both the *base* (139M parameters) and *large* (435M parameters) versions of ExtEnD presented in the reference work, using Rectified Adam as our optimizer, with 10^{-5} learning rate, and training with a batch size of 8000 tokens. All the trainings are done for a single run on GeForce RTX 3090 graphic card with 24 gigabytes of VRAM. Henceforth, we refer to these systems as GENRE^{def}, $\text{ExtEnD}^{def}_{base}$ and $\text{ExtEnD}^{def}_{large}$, respectively.

3.3 Results

In Table 2 we show the *inKB Micro* F_1 score that our models and their comparison systems achieve on the datasets under consideration. As a first note, we point out that, for easier comparability between our experiments, we reproduce the original AIDA-CoNLL models of both De Cao et al. (2021) and Barba et al. (2022). While we attain comparable performances for the latter, and hence omit it, we find that our own implementation of GENRE, which we denote in Table 2 by $\text{GENRE}^{\dagger}$, obtains better results than its reference, especially out of domain, with an average improvement of more than 2 points.

Moving to GENRE^{def}, its behavior is definitely below its counterpart with Wikipedia titles (i.e.,

[1] These candidate sets were generated through count statistics from Wikipedia, YAGO and a large Web corpus.

[2] We took the latest dump (June 13th, 2022) at the moment of writing from the official Wikidata website: https://dumps.wikimedia.org/wikidatawiki/entities/

	Model	In-domain		Out-of-domain					Avgs	
		AIDA$_{dev}$	AIDA$_{test}$	MSNBC	AQUAINT	ACE2004	CWEB	WIKI	Avg	Avg$_{OOD}$
AIDA+	Yang et al. (2018)	-	**95.9**	92.6	89.9	88.5	**81.8**	79.2	88.0	86.4
	GENRE	-	93.3	94.3	89.9	90.1	77.3	87.4	88.8	87.8
	ExtEnD$_{large}$	-	92.6	**94.7**	**91.6**	**91.8**	77.7	**88.8**	**89.5**	**88.9**
AIDA	GENRE	-	88.6	88.1	77.1	82.3	71.9	71.7	79.5	78.2
	ExtEnD$_{base}$	-	87.9	92.6	84.5	**89.8**	74.8	74.9	84.1	83.3
	ExtEnD$_{large}$	-	90.0	**94.5**	**87.9**	88.9	**76.6**	76.7	**85.8**	**84.9**
	GENRE†	94.8	90.7	91.3	76.9	87.3	73.9	73.7	82.3	80.6
	GENRE^{def}	93.2	84.4	83.1	59.6	81.3	64.0	63.4	72.6	70.3
	ExtEnD$^{def}_{base}$	93.9	89.1	93.5	84.9	87.7	74.9	74.5	84.1	83.1
	ExtEnD$^{def}_{large}$	**94.9**	**92.4**	93.2	87.0	87.7	76.4	**78.3**	**85.8**	84.5

Table 2: *inKB Micro* F_1 scores over the AIDA-CoNLL validation and test splits, and the out-of-domain datasets when training on AIDA-CoNLL (bottom), or on additional resources as well (top). The best score in each section is marked in **bold** and, in the bottom part, if its difference from its best alternative is statistically significant ($p < 0.01$ according to the McNemar's test (Dietterich, 1998)), we also underline it.

	Model	MFC	LFC	UE	UEM	UM
AIDA	ExtEnD$_{large}$	**98.3**	**81.6**	80.9	80.9	89.0
	ExtEnD$^{def}_{large}$	**98.3**	81.0	**86.9**	**86.5**	**92.9**
OOD	ExtEnD$_{large}$	**97.2**	82.2	73.8	74.4	77.2
	ExtEnD$^{def}_{large}$	96.5	81.5	**74.5**	**75.0**	**77.7**

Table 3: Fine-grained results analysis over the AIDA-CoNLL (top) and out-of-domain (bottom) datasets. Left to right, columns are Most Frequent Class (MFC), Less Frequent Class (LFC), Unseen Entity (UE), Unseen Entity-Mention pair (UEM) and Unseen Mention (UM). **Bold** and underline have the same meaning as in Table 2.

GENRE and GENRE†), with a drop of roughly 10 points on average. To better understand this issue, we analyzed its predictions over the validation set, but did not identify any significant error pattern. In particular, we investigated whether GENRE^{def} presented length biases or was excessively skewed towards the most frequent entities and, consequently, less apt to scale over less frequent entities or unseen mentions. Interestingly, we did not find either of these to be the case, with the two systems having similar error distributions. We believe instead that the drop might be happening as the formulation behind GENRE^{def} requires modeling a much more complex output space and more data could be needed to scale properly.

Considering, instead, extractive formulations, we find the role of definitions to be definitely more impactful. ExtEnD$^{def}_{base}$ surpasses ExtEnD$_{base}$ on 3 out of 5 out-of-domain benchmarks and on the standard test set, here by more than 1 point. However, arguably our most interesting finding is the behavior of ExtEnD$^{def}_{large}$. This system attains large statistically significant improvements on AIDA$_{test}$ (+2.4) and WIKI (+1.5) and comparable performances on CWEB.

Yet, when considering Avg and Avg$_{\text{OOD}}$, ExtEnD$^{def}_{large}$ appears to behave worse than its title-only alternative, with identical Avg and inferior Avg$_{\text{OOD}}$ performances. We argue that this is an unfortunate limitation of these two metrics, inherent to their nature of macro averages, and that statistical testing depicts a more complete landscape. On the one hand, MSNBC, AQUAINT and ACE2004 are all very small datasets (Table 1) where the apparently large performance drop between ExtEnD$^{def}_{large}$ and ExtEnD$_{large}$ is not statistically significant but rather caused by a few different classifications; to put things into perspective, on ACE2004, despite the 1.2 difference in F_1 score, the predictions of the two systems differ for a total of only 8 samples, with 5 and 2 being the number of these that only ExtEnD$_{large}$ and ExtEnD$^{def}_{large}$ get right, respectively. On the other hand, on the three remaining datasets – which are far larger – ExtEnD$^{def}_{large}$ either reports a statistically significant improvement (AIDA$_{test}$ and WIKI) or performs on par (CWEB), highlighting the benefits of our more expressive textual representations.

Finally, to further examine the impact of our proposal, we investigate the effectiveness of ExtEnD$^{def}_{large}$ over different classes of mention and label frequency, both in domain, i.e., over the test

set in AIDA-CoNLL, and out of domain, i.e., over the concatenation of the five datasets, and compare it with ExtEnD$_{large}$ (Table 3). Specifically, we consider instances:

- tagged with their most frequent entity in the training set (MFC);
- tagged with a less frequent entity (LFC);
- tagged with an unseen entity (UE);
- whose (mention, entity) pair does not appear in the training set (UEM);
- whose mention does not appear in the training set (UM).

Overall, apart from the MFC and LFC classes, where the difference is not statistically significant, ExtEnD$^{def}_{large}$ fares better in all other settings, which all require scaling over unseen patterns. Most notably, it yields $+6.0$ (AIDA) and $+0.7$ (OOD) improvements, both statistically significant, on unseen entities. This further underlines the better generalization capability granted by the use of more expressive textual representations.

4 Qualitative Analysis

We report in Table 4 a selection of examples from the WIKI dataset, showing candidates with both their title-only textual representations and those produced by our proposal. What we can see is that using the sole titles can result in imposing strong assumptions on what knowledge was captured by the model under consideration during its pre-training stage. For instance, in the first example in Table 4, the model needs to know, beforehand, that *Leed Rhinos* is an English rugby league football club. Moreover, relying only on titles can also result in underspecified queries. In the second example, if we were to look only at the titles provided for the two candidates, both alternatives would arguably be equally correct. A similar issue holds for the third example: although it may appear that the system could guess that the most likely candidate is the first one, as the second alternative is explicitly stated to be in Massachusetts, this strategy does not hold when considering the actual full list of candidates, which is not reported in Table 4 due to space constraints. While the model might be able to correctly predict these instances thanks to spurious correlations in the training set (e.g., the entity *Superman* being always linked to the superhero meaning, while *Superman (comic book)* to the meaning of comic book series), Table 3 clearly shows that this strategy does not scale.

Sentence: *Hugh Waddell is a Scottish [...] professional rugby league footballer [...] has played [...] at club level for [...] Leeds [...]* **Previous Candidates:** ✘ Leeds ✓ Leeds Rhinos **New Candidates:** ✘ Leeds: city in West Yorkshire, England ✓ Leeds Rhinos: English rugby league football club
Sentence: *World Without Superman is a Superman comic book story arc published by DC Comics.* **Previous Candidates:** ✓ Superman ✘ Superman (comic book) **New Candidates:** ✓ Superman: superhero appearing in DC Comics ✘ Superman: comic book series featuring Superman
Sentence: *Frank Mortimer born [...] in Wakefield was an English professional rugby league footballer [...]* **Previous Candidates:** ✓ Wakefield ✘ Wakefield, Massachusetts **New Candidates:** ✓ Wakefield: city in West Yorkshire, England ✘ Wakefield, Massachusetts: town in Massachusetts

Table 4: Extracts from the WIKI dataset, showing candidates with both the textual representations relying only on Wikipedia titles (**Previous candidates**), and our description-enhanced proposal (**New Candidates**). Due to space limitations, out of the 100 candidates all these three examples have, we only report the first two, which always include the correct one (denoted by ✓, as opposed to the incorrect alternative marked by ✘).

5 Conclusion

In this work, we focus on a shortcoming of generative and extractive formulations in Entity Disambiguation, namely their usage of Wikipedia titles, which are often insufficiently informative, and explore the effect of more expressive representations on these formulations. While we do not witness positive gains for generative formulations, at least in the limited data and computational regime we consider, we report strong improvements on extractive formulations. Specifically, our extractive approach sets a new state of the art on 2 out of the 6 benchmarks under consideration, and, more interestingly, shows better scalability over unseen patterns, especially unseen entities.

Limitations

We believe that our work has four major limitations. First, both the generative and extractive formulations that we consider lack parallelism, as they disambiguate each mention in the input text one at a time. While batching can definitely help, it poses additional computational requirements and, what is more, the same (but for the position of the <*s*> and </*s*> special symbols) input text would still need to be encoded multiple times. Second, our representation strategy requires the availability of descriptions in the target language in Wikidata (or some other knowledge base with a mapping from Wikipedia titles). While this data is readily available for English, this might not be the case for several other mid-to-low-resource languages. Third, both our formulations are local and, granted that pre-trained language models have certainly bridged the gap with global alternatives, their underlying independence assumption is still limiting. Finally, our proposal does incur an increased computational cost, with the textual representations getting considerably longer: while using Wikipedia titles results in sequences with an average subword length over AIDA-CoNLL of 7 and a 99th percentile of 14, adding descriptions nearly doubles the average, reaching 12.5, and makes the 99th percentile hit 29.

Acknowledgments

The authors gratefully acknowledge the support of the ERC Consolidator Grant MOUSSE No. 726487 under the European Union's Horizon 2020 research and innovation programme, and the CREATIVE project (CRoss-modal understanding and gEnerATIon of Visual and tExtual content) funded by the MIUR Progetti di ricerca di Rilevante Interesse Nazionale programme (PRIN 2020).

References

Edoardo Barba, Luigi Procopio, and Roberto Navigli. 2022. ExtEnD: Extractive entity disambiguation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2478–2488, Dublin, Ireland. Association for Computational Linguistics.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer.

Michele Bevilacqua, Rexhina Blloshmi, and Roberto Navigli. 2021. One SPRING to rule them both: Symmetric AMR semantic parsing and generation without a complex pipeline. In *Proceedings of AAAI*.

Samuel Broscheit. 2019. Investigating entity knowledge in BERT with simple neural end-to-end entity linking. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 677–685, Hong Kong, China. Association for Computational Linguistics.

Razvan Bunescu and Marius Paşca. 2006. Using encyclopedic knowledge for named entity disambiguation. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, pages 9–16, Trento, Italy. Association for Computational Linguistics.

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. 2009. *Introduction to Algorithms, 3rd Edition*. MIT Press.

Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2021. Autoregressive entity retrieval. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Thomas G Dietterich. 1998. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural computation*, 10(7):1895–1923.

Gabrilovich Evgeniy, Ringgaard Michael, and Subramanya Amarnag. 2013. FACC1: Freebase annotation of clueweb corpora, version 1 (release date 2013-06- 26, format version 1, correction level 0).

Farima Fatahi Bayat, Nikita Bhutani, and H. Jagadish. 2022. CompactIE: Compact facts in open information extraction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 900–910, Seattle, United States. Association for Computational Linguistics.

Stephen Guo, Ming-Wei Chang, and Emre Kiciman. 2013. To link or not to link? a study on end-to-end tweet entity linking. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1020–1030, Atlanta, Georgia. Association for Computational Linguistics.

Zhaochen Guo and Denilson Barbosa. 2018. Robust named entity disambiguation with random walks. *Semantic Web*, 9(4):459–479.

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.

Heng Ji and Ralph Grishman. 2011. Knowledge base population: Successful approaches and challenges. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1148–1158, Portland, Oregon, USA. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Phong Le and Ivan Titov. 2018. Improving entity linking by modeling latent relations between mentions. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1595–1604, Melbourne, Australia. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Andrea Moro, Alessandro Raganato, and Roberto Navigli. 2014. Entity linking meets word sense disambiguation: a unified approach. *Transactions of the Association for Computational Linguistics*, 2:231–244.

Luigi Procopio, Rocco Tripodi, and Roberto Navigli. 2021. SGL: Speaking the graph languages of semantic parsing via multilingual translation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 325–337, Online. Association for Computational Linguistics.

Hamed Shahbazi, Xiaoli Z. Fern, Reza Ghaeini, Rasha Obeidat, and Prasad Tadepalli. 2019. Entity-aware elmo: Learning contextual entity representation for entity disambiguation. *CoRR*, abs/1908.05762.

Simone Tedeschi, Simone Conia, Francesco Cecconi, and Roberto Navigli. 2021. Named Entity Recognition for Entity Linking: What works and what's next. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2584–2596, Punta Cana, Dominican Republic. 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.

Yi Yang, Ozan Irsoy, and Kazi Shefaet Rahman. 2018. Collective entity disambiguation with structured gradient tree boosting. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 777–786, New Orleans, Louisiana. Association for Computational Linguistics.

Wenpeng Yin, Mo Yu, Bing Xiang, Bowen Zhou, and Hinrich Schütze. 2016. Simple question answering by attentive convolutional neural network. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1746–1756, Osaka, Japan. The COLING 2016 Organizing Committee.