Investigating SoTA Entity-Linking Methods for Dialogue

Stanford CS224N Custom Project

Katherine Yu
Department of Computer Science
Stanford University
katherineyu@stanford.edu

Isaac Dan Zhao
Department of Computer Science
Stanford University
ikezhao@stanford.edu

Arpit Ranasaria
Department of Computer Science
Stanford University
arpitr@stanford.edu

Abstract

Our goal is to investigate how current SoTA entity linkers perform in the context of dialogue, and to compare the performance of different approaches to entity linking within this domain. We benchmark existing entity linkers on a conversational entity linking dataset, and develop our own entity linking model for the purpose of determining what methods for encoding and scoring entities result in better entity linking performance for dialogue. We base our model on ReFinED (Ayoola et al., 2022), which uses a bi-encoder to score mention and entity pairs. Due to the large scale of data and training time typically needed for entity linking model, we do not aim to achieve SoTA performance with our own model but rather investigate the effects of variations to its encoder. We find that our mention detection module results in a significantly higher recall on the ConEL-2 dataset than entity linkers that have primarily been trained on Wikipedia data and that encoding entities primarily by their KB labels results in higher entity disambiguation accuracy.

1 Key Information to include

- External collaborators : None
- External mentor : Ryan A. Chi (ryanchi@cs.stanford.edu)
- Sharing project: No

2 Introduction

In the domain of dialogue, we consider entity linking (EL) to consist of two subtasks, mention detection (MD), which is determining which tokens in a document refer to an entity, and entity disambiguation (ED), which is determining what entity a mention span corresponds to—entities are taken from a knowledge base (KB) such as Wikidata. Compared to the objective of most SoTA entity linkers, our first subtask differs in that we would like to detect concepts in addition to named entities. The task of identifying named entities (e.g. people, organizations, locations referred to by name in the given text) is referred to as named entity recognition (NER), and this is ineffective for conversational EL as it is common for speakers in a conversation to focus on concepts or more general nouns.

We are interested in determining an approach to entity linking most conducive for use by a social chatbot such as the open-domain dialogue agent Chirpy Cardinal developed by Chi et al. (2021). Current SoTA entity linkers such as ReFinED (Ayoola et al., 2022) and REL (van Hulst et al., 2020)

are pre-trained on Wikipedia data or news articles and generally aim to link proper nouns (e.g. "Berlin") to entities. Consequently, they achieve low recall on dialogue, missing entities from a speaker's utterance that could be used to determine the current topic of the conversation. Examples of this occurring on the ConEL-2 conversational dataset (Joko and Hasibi, 2022) are shown as follows:

Utterance	Entities	ReFinED	REL
SYSTEM: "Heavy metal is sooo good, and I'm not talking about something like lead either. It's a genre of rock music!" **USER:** "Ha! I'm familiar. It's not my first choice, but I do like some classic Led Zeppelin or Black Sabbath. How about you?"	"classic" → Classic rock "Led Zeppelin" → Led Zeppelin "Black Sabbath" → Black Sabbath	"Led Zeppelin" → Led Zeppelin "Black Sabbath" → Black Sabbath	"Led Zeppelin" → Led Zeppelin "Black Sabbath" → Black Sabbath
SYSTEM: "It feels like it was definitely from America but rock music was made in the UK back in the 1960's and 1970s." **USER:** "I could see that! On the heels of psychedelic music....all hail the electric guitar!	"psychedelic music" → Psychedelic music "electric guitar" → Electric guitar	None	None

Table 1: Entity linking results of ReFinED and REL on an example from the ConEL-2 dataset. Only user utterances are linked.

While these EL models do not achieve satisfactory performance in mention detection, they are precise in the ED stage, linking most detected entities correctly. Recognizing the disparity between MD and ED performance on dialogue, we develop separate MD and ED modules for our model, to allow them to be trained and analyzed separately. The MD module is trained directly on ConEL-2 while the ED module is both pre-trained on Wikipedia data and fine-tuned on ConEL-2. We find that the MD module significantly improves recall and pre-training on Wikipedia data boosts the accuracy of the ED module.

3 Related Work

In general terms, EL is an important aspect of decoding the meaning of unstructured text and is useful in extracting information that is most useful to a given task. Applications of EL include document annotation and KB population, which involves taking unstructured text and determining the entries that can be added to the KB. The ReFinED model is deployed on the latter task in Ayoola et al. (2022), specifically on a large (1 billion) set of web pages, and so the authors emphasize computational efficiency and zero-shot capability. For the purposes of a dialogue agent, we would also take these factors into consideration, as users usually expect low latency and may refer to entities not found in the training data. Due to these overlapping needs, we choose to take a similar approach for our ED module as in ReFinED.

In scoring mention and entity pairs, Ayoola et al. (2022) use a bi-encoder architecture in which the contextualized mention embedding and entity description embedding are computed separately and then used to compute a score. However, in Wu et al. (2020), the authors note that their cross-encoder, which concatenates the mention in context with the entity description before computing an embedding, outperforms their bi-encoder since the former can capture more interactions between the context and description. However, the cross-encoder requires more compute time and is not suitable for tasks requiring fast inference.

Besides mention and entity pairs, van Hulst et al. (2020) also consider the coherence of entity linking decisions with each other for a given document. In their REL model, each pair of entities is scored by combining their embeddings multiplicatively along with a normalization score based on the mention and context embeddings associated with each entity. The authors focus on ED and use an existing NER tool (Flair) for MD, which results in a modular architecture that allows for the use of different tools for MD. For our model, we also make use of a modular architecture.

A recent approach to conversational entity linking by Joko and Hasibi (2022) uses existing architectures for MD and ED fine-tuned on their ConEL-2 dataset to produce their model, CREL, which is an extension of REL. For MD, they employ a BERT-based model to classify tokens as beginning, inside, or outside mentions and for ED they use the same approach as in the REL model. We follow their approach for MD but develop our own module for ED, which allows us to evaluate different ED approaches.

4 Approach

4.1 Problem Formulation

Our aim is to improve MD performance and compare ED approaches on the ConEL-2 dataset. For the MD module, the task is as follows: given a sequence of tokens $X = \left[x_1, x_2, \ldots, x_{|X|}\right]$ in an utterance, create a function $\mathcal{X} : X \rightarrow \{B, I, O\}$ which classifies each token as beginning, inside, or outside a mention, generating a set $M = \left\{m_1, m_2, \ldots m_{|M|}\right\}$ of entity mentions. For the ED module, the task is as follows: given a KB with a set of entities $E = \left\{e_1, e_2, \ldots, e_{|E|}\right\}$ and a sequence of tokens $X = \left[x_1, x_2, \ldots, x_{|X|}\right]$ in an utterance with mentions $M = \left\{m_1, m_2, \ldots m_{|M|}\right\}$, create a function $\mathcal{M} : M \rightarrow E$ that assigns each mention to the correct entity. Note that although the utterance text is already passed into the MD module, it is used again to in order to generate contexualized mention embeddings that are passed into the ED module.

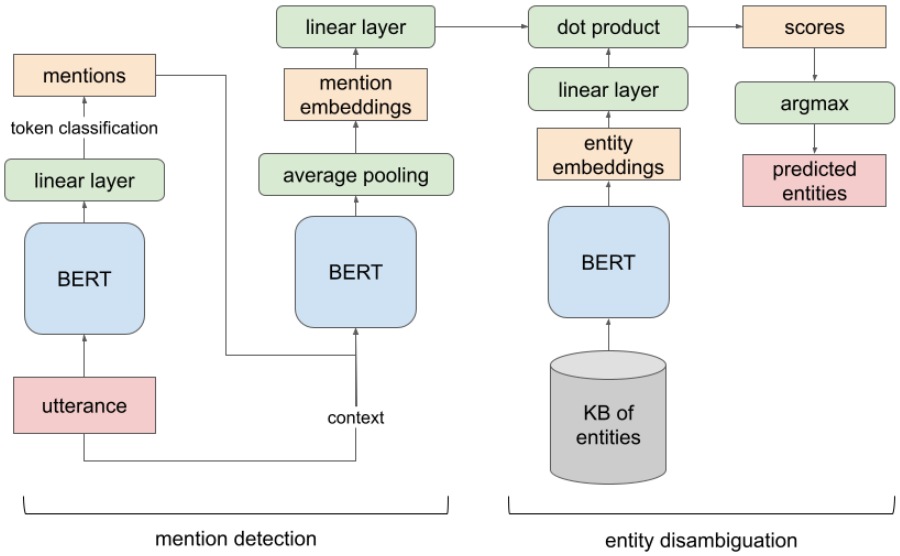

Figure 1: Architecture of our model. Word embeddings are computed using the 'bert-base-uncased' variant of the BERT model. In the MD module, the mentions are detected based on token classification and contextualized embeddings of the mentions are computed. In the ED module, a bi-encoder takes in precomputed entity embeddings and mention embeddings to compute a score for each mention and entity, and the entity with the highest score for each mention is chosen.

4.2 Mention Detection

For the MD module, we utilize the BertTokenizerFast transformer to tokenize inputs and fine-tune the BertForTokenClassification model by HuggingFace, which is a BERT model with a linear layer that takes in its hidden states output (Devlin et al., 2019). The model outputs the indices of the detected mentions in the original input as well as spans surrounding each mention with an average length of 64 characters (the context window is expanded slightly according to where the input is split by the tokenizer). As in Ayoola et al. (2022), we encode the detected mentions in a single forward pass for greater efficiency in the ED stage; each mention embedding is a vector $\mathbf{m_i} \in \mathbb{R}^{768}$. This also

makes our approach incompatible with a cross-encoder architecture that would require computing embeddings of mentions concatenated with entities.

4.3 Entity Disambiguation

We implement an ED module from scratch using Pytorch and following the approach of Ayoola et al. (2022). Similarly to the MD module, we precompute embeddings for each of the entities in our KB for faster inference. The embeddings are computed by passing in entity representations into a BERT model and taking the final layer embedding for the [CLS] token inserted at the beginning of the representation; each entity embedding is a vector $\mathbf{e_j} \in \mathbb{R}^{768}$. We considered three methods of representing entities: the embedding of a concatenation of the entity's Wikidata label and description, the embedding of the entity's Wikidata label, and a concatenation of the entity's Wikidata label embedding with an entity prior precomputed from our KB (in the below, f is the embedding function).

$$\text{Representation 1: } [f(\text{[CLS] label [SEP] description [SEP]})]$$
$$\text{Representation 2: } [f(\text{[CLS] label [SEP]})]$$
$$\text{Representation 3: } [f(\text{[CLS] label [SEP]}); P(\text{entity})]$$

We used a simple prior that was computed based on the "popularity" of each entity in our KB, specifically the number of page views for an entity's Wikipedia page accumulated over the time frame in which our dataset was collected. Representation 1 directly follows from the approach by Ayoola et al. (2022) while the other representations are chosen from experimenting on the ConEL-2 dataset before training on a larger corpus.

To compute a score for each mention and entity pair, we project the mention and entity embeddings into a shared vector space of dimension d using a linear layer for each and take the dot product of the projections. Our model trains for the optimal values of these layers.

$$s\left(m_i, e_j\right) = \left(A_1 \mathbf{m_i} + b_1\right) \cdot \left(A_2 \mathbf{e_j} + b_2\right)$$

Equation 1: Entity description score s, for mention m_i and entity e_j. where e_j is an entity. A linear layer is applied to each of the embeddings $\mathbf{m_i}$ and $\mathbf{e_j}$.

We train the ED module using cross-entropy loss on the softmax distribution computed from the vector of scores $\mathbf{s} \in \mathbb{R}^{|E|}$ for a mention m and its gold label, which we represent as a one-hot vector $\mathbf{g} \in \mathbb{R}^{|E|}$, where $|E|$ is the number of entities in our KB.

$$\mathcal{L}_{CE}(m) = -\sum_i \mathbf{g_i} \log(\mathbf{s_i})$$

Equation 2: Cross-entropy loss for a mention m with entity scores $\mathbf{s}$ and gold label $\mathbf{g}$.

4.4 Baseline

For our baseline model, we implement the ED module with representation 1 as described for entity embeddings and train it on the ConEL-2 dataset. We also benchmark ReFinED (Ayoola et al., 2022), REL (van Hulst et al., 2020), and CREL (Joko and Hasibi, 2022) on this dataset, expecting low recall from the first two entity linkers, which were developed for NER.

5 Experiments

5.1 Data

The dataset that we used for benchmarking and fine-tuning is the ConEl-2 dataset (Joko and Hasibi, 2022), created by annotating conversations from the Wizard of Wikipedia (WoW) (Dinan et al., 2018) dataset, which contains multi-turn conversations between two human speakers, a "wizard" and "apprentice." During the data collection process for WoW, the "wizard" is instructed to form

utterances based on passages from Wikipedia that are relevant to the current conversation topic while the "apprentice" is instructed to talk freely and play the role of someone who is learning about the topic. This results in conversations that have enough depth to be conducive to entity linking. In ConEL-2, the provided annotations are the gold labels for the entities that should be linked in each utterance. The authors also study personal entity linking and include annotations for spans referring to personal entities as well. Overall, the dataset contains 290 conversations, each of which consists of about 5 to 10 utterances. We processed the ConEL-2 dataset into files suitable for MD and ED.

A limitation of the ConEL-2 dataset is its small size relative to other entity linking datasets. To account for this, we also use the Kensho Derived Wikimedia Dataset [1], which contains preprocessed subsets of Wikipedia and Wikidata dumps collected at the end of 2019. We used the list of Wikipedia pages in the dataset to choose the top 200,000 entities by number of page views for our KB (for each mention, our ED module chooses from among these 200,000 entities to link). For each entity, we used its Wikidata label and description in the dataset to precompute an embedding. The dataset also includes Wikipedia pages in plaintext with annotations of spans that are linked to other pages. We use this dataset to pre-train our ED module by treating the targets of the links as gold labels for the spans.

5.2 Evaluation method

We evaluated all models on the test set provided in the ConEl-2 dataset, using precision, recall, and F1 score as the metrics. A true positive is a correctly linked entity, a false positive is a linked entity that is not found in the gold labels, and a false negative is an entity in the gold labels but not predicted. For the purposes of entity linking, we do not consider true negatives since these are not well-defined by the data; this also matches the evaluation approach of Joko and Hasibi (2022).

5.3 Experimental details

We trained our models on an NVIDIA A10G Tensor Core GPU with 8 vCPUs.

For the MD module, we performed a hyperparameter search on the optimizer and learning rate, finding that an SGD optimizer, learning rate of 5e-3, and 5 epochs led to the best performance. We trained the MD module only on the ConEL-2 dataset, since other datasets such as Wikipedia would result in low recall scores. Due to the small size of the dataset, the training time was only a few minutes for each configuration we evaluated.

Taking into account that the ReFinED and REL models would achieve low recall on a conversational entity linking dataset, we also benchmarked a combination of our MD module with each of these models for the ED stage. This was done by using our MD module to extract mentions from the input and passing them in to the other model, as we noted that given a shorter input these entity linkers are more likely to predict entities even if they are not proper nouns (whereas in longer inputs these entities would be ignored).

We pre-trained our ED module on a random sample of the Wikipedia pages in the Kensho Derived Wikimedia Dataset (to reduce training time). For the hyperparameters, we found that using an AdamW optimizer with a learning rate of 1e-5 resulted in better performance. For each of the entity representations described in our approach, we trained the ED module for 5 epochs on the sample of Wikipedia pages, which took 3 hours each time. After pre-training on the Wikipedia data, we fine-tuned the ED module on the ConEL-2 dataset. For fine-tuning, we used an AdamW optimizer with a learning rate of 1e-4 and trained for 10 epochs.

5.4 Results

Mention Detection Recall

Our MD module achieved a precision score of 0.818, recall score of 0.779, and F1 score of 0.798. While we could not benchmark MD for the other entity linkers we have discussed since they do not have standalone MD modules, we approximated a recall score for MD by summing the number of entities predicted for each utterance and dividing by the total number of entities present in the data. This would result in a recall score equal to or higher than the true recall score since wrongly predicted mention are also counted.

[1] `https://www.kaggle.com/kenshoresearch/kensho-derived-wikimedia-data`

Model	Recall
ReFinED	0.353*
REL	0.294*
CREL	0.765*
MD module	0.**779**

Table 2: Recall scores for MD on the ConEL-2 dataset. *Approximate.

Entity Linking Performance

As was expected due to the limitations of our compute resources and time frame, we were not able to sufficiently train our ED module to achieve a precision score comparable to that of current SoTA entity linkers (for comparison, Ayoola et al. (2022) pre-trained the ReFinED model for 48 hours using 4 GPUs and Wu et al. (2020) pre-trained the BLINK bi-encoder for 70 hours using 8 GPUs). However, we did achieve a notably higher recall score than these two models, which even resulted in a higher F1 score, although the CREL model still achieved the best performance by far.

Combining our MD module with ReFinED and REL resulted in improved recall and F1 scores, although precision decreased. This is to be expected as making more predictions would result in more false positives and our method of passing in detected mentions to these models by only including the spans directly corresponding to each mention removes their ability to make predictions based on context.

Model	Precision	Recall	F1
ReFinED	**0.897**	0.183	0.304
REL	0.706	0.169	0.273
CREL	0.659	**0.608**	**0.632**
MD module and ReFinED	0.724	0.387	0.505
MD module and REL	0.561	0.552	0.556
MD and ED modules	0.384	0.383	0.383

Table 3: EL performance on the ConEL-2 dataset

Entity representation and training data	Precision	Recall	F1
Rep. 1, ConEL-2	0.163	0.164	0.163
Rep. 1, Wikipedia	0.146	0.155	0.151
Rep. 1, Wikipedia and ConEL-2	0.292	0.296	0.294
Rep. 2, ConEL-2	0.262	0.263	0.263
Rep. 2, Wikipedia	0.281	0.289	0.285
Rep. 2, Wikipedia and ConEL-2	0.380	0.378	0.379
Rep. 3, ConEL-2	0.185	0.190	0.188
Rep. 3, Wikipedia	0.304	0.312	0.308
Rep. 3, Wikipedia and ConEL-2	**0.384**	**0.383**	**0.383**

Table 4: ED performance on the ConEL-2 dataset resulting from different entity representations and training data for our ED module.

We consider our main contribution to be the comparison of different entity representations' effects on performance for our ED module. Initially, we represented entities by taking the BERT model embedding of the concatenation of an entity's label and description from Wikidata, as in Ayoola et al. (2022). A similar representation is used by Wu et al. (2020), where the label and description are taken from Wikipedia. Surprisingly, we found that our ED module performed significantly better with the second entity representation we evaluated, which is the embedding of the entity's label only. We hypothesize that this is due to most mentions having a similar embedding to embeddings of only their Wikidata label, and concatenating the description weakens the similarity between the two embeddings. However, it is clear that incorporating some description of an entity in its representation is still necessary in order to disambiguate two entities that have almost identical labels. Our results suggest that perhaps entity embeddings should be weighted more heavily towards their labels in the KB in order to account for the similarity between mentions and labels.

Including a prior in the entity embedding reduced the performance of the ED module when trained only on ConEL-2, but resulted in better performance after pre-training on Wikipedia. From this, we can infer that the model learned to weigh the prior in its predictions more accurately from the Wikipedia data. In all cases, pre-training on Wikipedia and then fine-tuning on ConEL-2 led to a significantly better performance than only pre-training or only training on ConEl-2. In contrast, we also fine-tuned ReFinED on ConEL-2 using their provided fine-tuning script and saw almost no difference in performance, which could be due to the small size of the ConEL-2 relative to the pre-training dataset used for ReFinED.

6 Analysis

Error analysis

Of the correct predictions that our model makes, most are mentions that are referred to by a span that is the same or almost as its Wikidata label, which we would expect from representing entities with the embeddings of their labels. Among the incorrectly linked entities our model chooses, most are semantically related to the correct entity or have a similar label.

Utterance	Correct Entities	Predicted Entities
"I've always been interested in Extraterrestrial life - I wonder if any exists out there in space!"	"Extraterrestrial life" → Extraterrestrial life "space" → Outer space	"Extraterrestrial life" → Extraterrestrial life "space" → Solar system
"Which states use NCAA rules?"	"NCAA" → National Collegiate Athletic Association	"states" → Nation state "NCAA rules" → Host (biology)
"Without cheese? That is just an insult to pizza! Though I guess it is mostly for those who are lactose intolerant."	"cheese" → Cheese "pizza" → Pizza "lactose intolerant" → Lactose intolerance	"cheese" → Cheese "pizza" → Pizza "lactose into" → Calorie restriction
"Red is a very nice color. Vibrant and looks good on cars."	"Red" → Red "color" → Color "cars" → Car	"Red" → Red (Taylor Swift album) "color" → Color "cars" → Car

Table 5: Selected examples from the ConEL-2 test set.

In the last row of table 5, we can see that the mention of "Red" is incorrectly linked by our model to the Taylor Swift album of the same name. We first speculated that the entity corresponding to the Taylor Swift album appears in the Wikipedia pre-training data more often than the color red, but in fact the album never appears in the training data while the color red does. Thus, this mistake must be explained by the fact that the album has a higher page view count in our dataset than the color red. Indeed, inputting the same utterance to our model that does not take into account priors correctly links "Red" to the color. While including the prior did lead to a slightly better overall performance, we should be wary of the possible mistakes it can lead the model to make.

Another source of error in our model is that mistakes in the MD module propagate to the ED module. In some examples, the MD module detects partial words as entities, as in the third row of table 5, detecting "lactose into" instead of "lactose intolerant". Although we designed our model to have a modular architecture that allows for training the MD and ED modules separately, such errors might be lessened by also training them simultaneously on a suitable dataset.

Overfitting

While the ConEL-2 dataset covers a variety of topics in its conversations, there are some characteristics of the dataset that are specific to a text-based conversation and may not translate to dialogue agents that converse with users verbally. In the dataset, there are several instances of the abbreviation "LOL", which our model predicts accurately after fine-tuning on ConEL-2. Furthermore, almost all utterances have punctuation which may affect mention embeddings and there are occasional typos that would not occur in a verbal conversation.

Zero-shot capability

As in Ayoola et al. (2022) and Wu et al. (2020), our model is zero-shot-capable since it is trained on not only the target dataset, ConEL-2, but also KB data. While we did not evaluate our model on a zero-shot dataset, we were able to confirm this capability on a few manually chosen entities in our KB that are not in our training data.

Utterance	Predicted Entities
"Iguanas are fascinating"	"Iguanas" → Iguana
"Kamala Harris is the vice president"	"Kamala Harris" → Kamala Harris

Table 6: Zero-shot predictions.

7 Conclusion

We examine the performance of current SoTA entity linkers on a conversational dataset and find that they achieve low recall without fine-tuning on in-domain data, as most entity linkers only link proper nouns, which is not sufficient for dialogue agents to understand the topics that a user may mention. To address the issue of low recall, we develop an EL model with separate modules for MD and ED so that the MD module can be trained specifically on conversational data. For the task of MD, our module achieves higher recall than the existing entity linkers we benchmarked. With our ED module, we were unable to approach SoTA performance in end-to-end EL due to limitations of compute resources and training time, but the higher recall achieved due to our MD module did result in a higher F1 score than the entity linkers that were not fine-tuned on conversational data.

With our ED module, we investigate the effects of different entity representations and find that representing entities primarily by the embedding of their Wikidata labels leads to improved performance. However, we did not use embeddings learned specifically for the task of EL. For future work, we would consider using a different model for our embeddings, such as LUKE, Language Understanding with Knowledge-based Embeddings (Yamada et al., 2020), and design our model to take into account other entity characteristics, such as Wikidata properties.

References

Tom Ayoola, Shubhi Tyagi, Joseph Fisher, Christos Christodoulopoulos, and Andrea Pierleoni. 2022. ReFinED: An efficient zero-shot-capable approach to end-to-end entity linking. In *NAACL 2022*.

Ethan A. Chi, Caleb Chiam, Trenton Chang, Swee Kiat Lim, Chetanya Rastogi, Alexander Iyabor, Yutong He, Hari Sowrirajan, Avanika Narayan, Jillian Tang, Haojun Li, Ashwin Paranjape, and Christopher D. Manning. 2021. Neural, neural everywhere: Controlled generation meets scaffolded, structured dialogue. In *Alexa Prize SocialBot Grand Challenge 4 Proceedings*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2018. Wizard of wikipedia: Knowledge-powered conversational agents.

Johannes M. van Hulst, Faegheh Hasibi, Koen Dercksen, Krisztian Balog, and Arjen P. de Vries. 2020. Rel: An entity linker standing on the shoulders of giants. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 2197–2200, New York, NY, USA. Association for Computing Machinery.

Hideaki Joko and Faegheh Hasibi. 2022. Personal entity, concept, and named entity linking in conversations. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, CIKM '22, page 4099–4103, New York, NY, USA. Association for Computing Machinery.

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

Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. 2020. Luke: Deep contextualized entity representations with entity-aware self-attention.