PeerJ Computer Science

A data-centric way to improve entity linking in knowledge-based question answering

Shuo Liu, Gang Zhou, Yi Xia, Hao Wu and Zhufeng Li

Information Engineering University, Zhengzhou, Henan, China

ABSTRACT

Entity linking in knowledge-based question answering (KBQA) is intended to construct a mapping relation between a mention in a natural language question and an entity in the knowledge base. Most research in entity linking focuses on long text, but entity linking in open domain KBQA is more concerned with short text. Many recent models have tried to extract the features of raw data by adjusting the neural network structure. However, the models only perform well with several datasets. We therefore concentrate on the data rather than the model itself and created a model DME (Domain information Mining and Explicit expressing) to extract domain information from short text and append it to the data. The entity linking model will be enhanced by training with DME-processed data. Besides, we also developed a novel negative sampling approach to make the model more robust. We conducted experiments using the large Chinese open source benchmark KgCLUE to assess model performance with DME-processed data. The experiments showed that our approach can improve entity linking in the baseline models without the need to change their structure and our approach is demonstrably transferable to other datasets.

Subjects Artificial Intelligence, Data Mining and Machine Learning, Data Science

Keywords Entity linking, Negative sampling, Natural language processing, Knowledge-based question answering

Submitted 28 September 2022
Accepted 11 January 2023
Published 9 February 2023

Corresponding author
Zhufeng Li, 20086538@qq.com

Academic editor
Xiangjie Kong

Additional Information and Declarations can be found on page 16

DOI **10.7717/peerj-cs.1233**

INTRODUCTION

In order to make use of ever-increasing quantities of data, many knowledge bases (KBs), such as Freebase (*Bollacker et al., 2008*) and DBpedia (*Auer et al., 2007*), have collected natural language network data which they formatted as triples (h, r, t), where h and t are subject and object entities, and r is the relation between them. Regular internet users may lack of the technical skills required to query such datasets easily. KBQA aims to answer the natural language question revolves around the KB, which provide a portable way to access KB for normal users (*Huang et al., 2021*). In KBQA, entity linking (EL) is an important approach to connecting natural language queries to formalized KBs and is usually considered to be the first step in creating a KBQA system. Table 1 shows the functions of each subtask and Fig. 1 shows the pipeline of such a tripartite EL system. The *mention* (*Schindler et al., 2022*) of an entity, in the remainder of the article, is the set of representations in natural language that include the phrase identifying the entity.

How to cite this article Liu S, Zhou G, Xia Y, Wu H, Li Z. 2023. A data-centric way to improve entity linking in knowledge-based question answering. *PeerJ Comput. Sci.* **9**:e1233 http://doi.org/10.7717/peerj-cs.1233

Table 1 Definition of the subtasks.

Subtask	Definition
Mention Detection	Identify entity mention from a question.
Candidate Generation	Produce a set of candidate entities in KG from the ambiguous entity mention
Entity Ranking	Rank candidate entities according to the mention-entity correspondence estimation

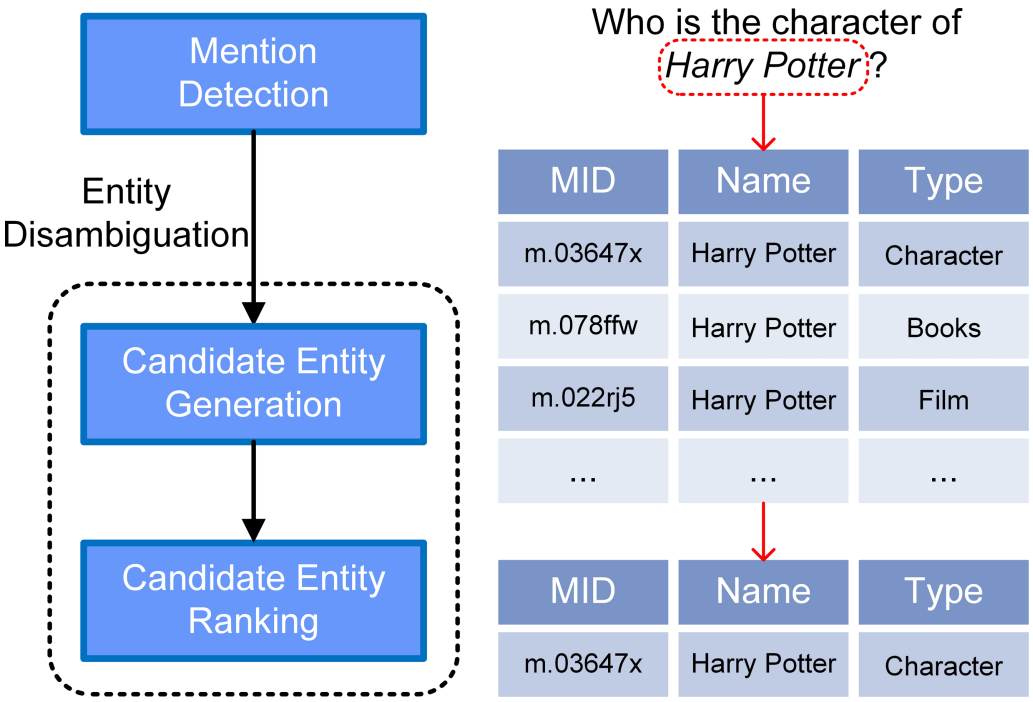

Figure 1 **A specific example of our pipeline model.** The input question is "Who is the character of Harry Potter?" Typical results of the query are shown. And our model successfully linked the mention "Harry Potter" to the correct entity node of the KB.

Full-size DOI: 10.7717/peerjcs.1233/fig-1

In recent research, there has been a tendency to transform the mention detection task into a sequence recognition task, and this technique has resulted in excellent performance of recurrent neural networks models, such as LSTM-CRF (*Lample et al., 2016*). EL concentrates on disambiguating entities, both candidate entity generation and candidate entity ranking are based on similarity. To improve EL, we determined it was necessary to better represent mention and entity. The semantics of a *mention* is usually influenced by the context, so polysemy is an extremely common phenomenon in natural language. Representing the mentions and entities in a multi-dimensional way is significant in entity disambiguation. Recent research in short text EL tend to enhance the presentation of mentions and entities through some external information such as the context of the mentions and the neighboring relations of the target entity. However, this approach has two drawbacks. Firstly, the relevant work has usually been

model-centric and has tended to improve the model only using several datasets rather than more generally. Secondly, most Chinese KBs are immature and therefore provide only limited information, making it necessary to somehow incorporate additional information to improve the performance of the EL model. Moreover, generating negative samples is necessary for training. Conventional negative sampling is usually based on random or normal distributions, but a model cannot learn enough valuable information due to the poor fitting provided by a straightforward statistical method.

We therefore developed a model, DME (Domain information Mining and Explicit expressing), to better representing mentions without changing the model structure. To improve the quality of negative samples, our approach to negative sampling combines surface form with semantic information of the entities.

In summary, the main contributions of this study are as follows.
(1) We increased the robustness of EL in the model without changing the model structure by designing a data-centric model, DME, to mine domain information for short text.
(2) We developed an innovative negative sampling approach that generates high quality negative samples by considering both surface form and semantic information.
(3) We performed experiments using KgCLUE and NLPCC2016 and thus demonstrated our methods are effective and adaptable.

RELATED WORK

Sevgili et al. (2022) and *Shen, Wang & Han (2015)* have conducted a detailed survey about the approaches of entity linking. Much research into entity linking is concerned with entity disambiguation, in which the key issue is how best to represent the semantics of the mention and the entity and how then to rank candidate entities by semantic distance. To increase efficiency, in terms of minimizing computing time, an inexact matching may be determined that can be prioritized over deep semantic matching, which allows entity disambiguation to be divided into processes of candidate entity generation and entity ranking.

Candidate entities generation

One approach is to match surface forms, which generates a candidate entity list by matching the surface forms between mention and entity. Many heuristics, such as Levenshtein distance (*Le & Titov, 2019*), n-grams (*Moreno et al., 2017*) and Word2vec (*Zwicklbauer, Seifert & Granitzer, 2016*), are used for embedding and matching the surface form of mention and entity. An alternative approach is to build an entity-mention dictionary that is expanded with aliases. Most aliases are extracted by KG metadata, such as entity pages, redirect pages, disambiguation pages and hyperlinks in Wikipedia articles (*Zwicklbauer, Seifert & Granitzer, 2016*; *Fang et al., 2019*).

Candidate entities ranking

Research to date has been primarily concerned with ranking candidate entities rather than inexact matching. The semantic matching model is optimized to capture the deeper semantics of both mention and entity. The two principal techniques used are context-mention encoding and entity encoding.

Context–mention encoding is intended to encode the mention with information captured from the context. The usual approach is to construct a dense contextualized vector that represents the mention. An earlier approach was to encode mentions using a convolutional neural network (*Francis-Landau, Durrett & Klein, 2016*). However, the mainstream approach now is to use recurrent neural networks with self-attention (*Vaswani et al., 2017*). Some researchers have used LSTM to build a recurrent neural network to improve representation (*Fang et al., 2019*; *Le & Titov, 2019*). For example, *Sil et al. (2018)* encoded mention contexts using LSTM and passed the result to a coreference chain and adjusted the representation using a tensor network. *Eshel et al. (2017)* modified LSTM-GRU by incorporating an attention mechanism into the encoder. Attention neural network is widely used in current encoding approaches. The entity linking model proposed in (*Logeswaran et al., 2019*; *Peters et al., 2019*; *Chen et al., 2020*; *Wu et al., 2020*) exploited pre-trained BERT to capture as many multidimensional features as possible to improve mention and entity encoding.

The second approach, entity encoding, is intended to capture deep semantic information and generate a distributed vector representation for each candidate entity. In earlier research, the entity representation space was populated with unstructured texts using algorithms, such as Word2vec, that produced co-occurrence statistics (*Zwicklbauer, Seifert & Granitzer, 2016*; *Moreno et al., 2017*). Recent work has used pretrained language models (PLMs), of which BERT (*Devlin et al., 2019*) is representative, to encode entities (*Nie et al., 2018*; *Mulang' et al., 2020*). For example, *Logeswaran et al. (2019)* and *Wu et al. (2020)* trained BERT using entity description pages from Wikipedia to form a supplementary representation of external information. *Yamada et al. (2019)* developed an entity disambiguation model which was trained using a novel masked entity prediction task. The model was trained by predicting randomly masked entities in entity-annotated texts from Wikipedia.

In addition to the approaches of encoding, some studies have improved the effectiveness of model training by enhancing the quality of training samples. To better training of the model, *Rao, He & Lin (2016)* exploited interactions in triplet inputs over the question paired with positive and negative examples. For the same purpose, *Zhang et al. (2019)* proposed a GAN-based methods, NSCaching, to sampling negative triplets from a KB.

Most of the existing approaches are modifications of existing models, which work better on certain data but are less adaptable. It means that they are not suitable for other datasets of the same task. In contrast to these approaches, our proposed DME model focuses on data-centric augmentation, and this model can simply but effectively improve the quality of text representation while being applicable to the vast majority datasets.

Task definition

A popular implementation of EL in KBQA is to use a pipeline consisting of two modules: *mention detection* and *entity disambiguation*. However, the huge volume of data in a KB will create a large search space when calculating mention–entity similarity. *Entity disambiguation* thus can be split into two subtasks, including *candidate generation* and *entity ranking* (*Sevgili et al., 2022*). $G = (V, E)$ represents the entities and relations in a KB, where

$V = \{v_1, v_2, \ldots, v_n\}$ contains all of the entities, $Q = \{q_1, q_2, \ldots, q_n\}$ and $M = \{m_1, m_2, \ldots, m_n\}$ are respectively the set of questions and the set of mentions that have been extracted from the questions. We can then describe the entity linking as: Given a set of entities V and a set of mentions M that are contained in a set of questions Q, entity linking model aims to link the mention $m \in M$ that appears in a question to the entity node $e \in V$ correctly. The task can be further formulated as:

$$\hat{a} = \underset{e \in V}{arg\ max} P_r(e|G, Q) \tag{1}$$

where $Pr(e|G, Q)$ is the probability of entity e being the correct linking result of the question Q. The target KB normally contains millions of entities, which means that directly modeling $Pr(e|G, Q)$ is computationally intensive. One line of research forms entity linking as a *semantic matching* task, aims to finds a suitable entity within KB that is similar to the *mention* in the question. Following this direction, we divided the entity linking into three steps: (1) Recognize the mention $m$from the question $q \in Q$, where m is the sub-string of q; (2) Extract the candidate entities $\overline{v}$ that match mention m andconstruct a set of candidate entities $\overline{V} = \{\overline{v}_p, \overline{v}_{p+1}, \ldots, \overline{v}_q\}, \overline{V} \subseteq V$; (3) Rank the candidate entities $\overline{V}$ to obtain the result v. The model can be factorized as:

$$\begin{aligned} &P_r(e|G, Q) = P_r\left(m, \overline{V}, v|G, Q\right) \\ &= P_m(m|G, Q) \cdot P_{\overline{V}}\left(\overline{V}|m, G, Q\right) \cdot P_v\left(v\left|\overline{V}, m, G, Q\right.\right) \end{aligned} \tag{2}$$

where $P_m(m|G, Q)$ is the mention detection model, $P_{\overline{V}}\left(\overline{V}|m, G, Q\right)$ is the model of candidate entity generation, and $P_v\left(v\left|\overline{V}, m, G, Q\right.\right)$ is a component for candidate entity ranking.

METHOD

We used a conventional approach for mention detection model $P_m(m|G, Q)$ and candidate entity generation model $P_{\overline{V}}\left(\overline{V}|m, G, Q\right)$. And the central goal of our research was to use the DME model to improve the performance of the candidate entity ranking model $P_v\left(v\left|\overline{V}, m, G, Q\right.\right)$.

Mention detection and candidate entities generation

The mention detection model $P_m(m|G, Q)$ detects the mention span within a question. Most recent approaches depend on named entity recognition (NER), which performs extremely well in mention detection. We considered mention detection to be a sequence labeling task and created a mention detection model using BERT and CRF.

We first transformed the question $q \in Q$ as a BIO-labeled sequence, where B and I represent the start and inner of a mention span, and O labels the superfluous part of q. We then used BERT and CRF to model the labeled sequence.

When the mention contained in the question had been obtained, we used $P_{\overline{V}}\left(\overline{V}|m, G, Q\right)$ to generate a set of candidate entities. Conventional approaches to candidate entity generation are mainly based on string comparison between the surface form of the mention entity and the name of the entity existing in a KB. We created a semantic space using a pretrained Word2vec algorithm and then encoded the mentions

and all entities in the KB for approximate similarity matching. We finally ranked the entities by the similarity score and returned the top-k entities as the list of candidate entities. The k is a variable and we found the best recall ratio when $k = 70$.

Candidate entities ranking

The last part of our entity linking model was to rank the candidate entities using the model $P_v\left(v\left|\overline{V}, m, G, Q\right.\right)$, which sorts candidate entities by deep semantic matching. It is essential when mining the deeper semantics of mentions and entities to increase the dimensions of the representations. A conventional approach is to learn more features by adjusting the structure of a model and varying its parameters. This approach requires researchers to prepare massive quantities of data to train models and to expect the deep neural network to automatically capture the inexplicable features. However, for existing giant models that had been derived from BERT, some hard problems demonstrated unexpected weaknesses in this approach when being used in downstream tasks. These weaknesses included elementary errors when treating the datasets in actual conditions and magnification of bias embedded in everyday human-originated questions. In short, improvements resulted in decreasing marginal benefits after the PLMs had been developed to a particular stage. Researchers are becoming increasingly aware of the importance of the quality of data. *Ng (2021)* suggested that research into artificial intelligence should change its approach from model-centric to data-centric. Also, *Lu et al. (2022)* recognized that keywords can better represent a sentence than use all information within a sentence. In our research, we found that the context of a mention and the relations that surround an entity can carry much information concerning the domain. For example, in the question "Who directed Game of Thrones?", the verb "direct" indicates that this question may be related to a film or television work. In the one-hop relations of the entity "Game of Thrones", "Release time", "Producer" and "Screenwriter" also indicate that the entity may be connected to a film or television work. These insights led us to realize that domain keywords are vital in entity linking. We devised a lightweight and transferable method of finding domain keywords by feature engineering and improve the performance of entity linking without modifying the structure of the model.

In this way, we developed the DME model to mine and explicitly represent the implicit domain information concealed in the text. Figure 2 shows the overall structure of DME. In detail, we used $A = \{a_1, a_2, \cdots, a_n\}$ to represent the feature words that consist of a question's segments or an entity node with surrounding relations. $P\left(B_i\left|a_j\right.\right)$ represents the implicit domain indication inherent in feature term $a_j$in domain B_i. To highlighting frequently used words, we modeled the occurrence frequency with the parameter popularity p_{a_j}:

$$p_{a_j} = \frac{df\left(a_j\right)}{\sum_{i=1}^{n} df\left(a_i\right)} \tag{3}$$

where the dictionary frequency $df\left(t\right)$ represents the number of domain directories that contains the feature term t. Then DME can be entirely represented as:

$$F\left(A\right) = selectS_{B_i} = select\left(\sum_{j=1}^{n} p_{a_j} \cdot P\left(B_i\left|a_j\right.\right)\right) \tag{4}$$

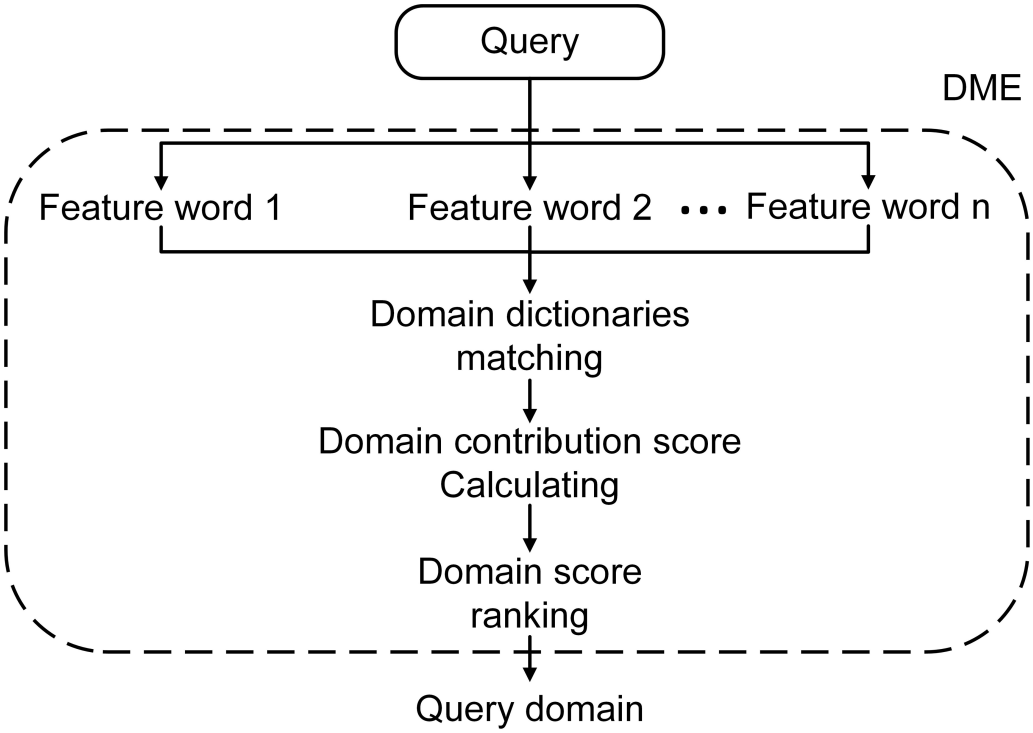

Figure 2 **The overall structure of DME.**
Full-size DOI: 10.7717/peerjcs.1233/fig-2

where S_{B_i} represents the different domain scores of A, and *Select* is the selection method we developed to process the real world's occurrence of a text being related to many different domains. We first sort S_{B_i} and obtain the standard deviation σ and then calculate the difference in score between the top-2, $s_1 = S_{B_1} - S_{B_2}$. If $s_1 > \sigma$, then B_1 will be returned as the result of $F(A)$. If $s_1 \leq \sigma$, both B_1 and B_2 will be regard as the result; S_{B_2} and S_{B_3} are then compared in the same manner.

Careful consideration needs to be given to the division of domains, as fewer domain divisions would make the distinction insufficient, while too many would make some words difficult to classify. We therefore learned from the Universal Decimal Classification (UDC) and some of its subsequent classification criteria (*McIlwaine, 1997*). The encyclopedia knowledge is therefore divided into nine domains. Table 2 gives examples of typical subdomains and the number of keywords for each branch. We constructed nine domain dictionaries based on millions of domain keywords.

Using Bayes' theorem, the local domain contribution $P\left(B_i \middle| a_j\right)$ can be modeled as:

$$P\left(B_i \middle| a_j\right) = \frac{P(B_i)P\left(a_j|B_i\right)}{P\left(a_j\right)} \propto P(B_i)P\left(a_j|B_i\right) \tag{5}$$

where $P\left(a_j\right)$ is the probability of a feature term occurrence, which can be treated as a constant. The probability of a randomly selected domain $P(B_i)$ is a prior probability:

$$P(B_i) = \frac{\log Count(B_i)}{\sum_{i=1}^{9} \log Count(B_i)}. \tag{6}$$

Table 2 Domain dictionary.

Domain	Sub-domains	Number
Nature	Organism, Natural resources, Astronomical phenomena	152,000
Culture	History, Literature, Historical personages, Religion	235,600
Daily	Diet, Traffic, Tourism, Entertainment	46,000
Social	Law, Organizations, Media, Charities	103,000
Technology	Computer, Medical and Vehicle technology	200,000
Art	Painting, Music, Opera and theatre, Movies and TV, Architecture	22,280
Sport	Series competitions, Electronic sports, Team names	23,000
Politics	Military affairs, Administrative divisions, Diplomacy	131,000
Economy	Enterprise, Brands, Stock and funds, Insurance	54,000

Since contribution is a relative concept that needs to be normalized later, it is not necessary to calculate the probability. We adapt the calculation in a uniform scale, which also improve the operational efficiency. We thus replaced $P\left(a_j|B_i\right)$ in (5) with $C\left(\mathrm{a}_j, B_i\right)$, which represents the contribution of feature term a_j to domain B_i:

$$C\left(\mathrm{a}_j, B_i\right) = \frac{1}{df\left(a_j\right)}. \tag{7}$$

Figure 3 shows an example of the DME model. For the question "What is the architectural style of Notre Dame de Paris?", a_1 and a_2 is the feature terms are obtained by matching the question with dictionaries. As shown in the Fig. 3, the standard deviation σ be calculated as 0.012. After calculating $s_1\,(0.008 \leq \sigma)$ and $s_2\,(0.02 > \sigma)$, the domain of the query $F\,(A)$ is Art and Culture. We can similarly obtain the entity domain by analyzing the relations around it *via* DME.

After obtaining the domain information of both mentions and entities as external information, we used it as a part of the input for training. In training, we represented each pair of a mention m and an entity $v$as a sequence in the format: "*mention#field#questionpattern*"and "*entityname#field#description*", where *question pattern* is the question string with the *mention* removed, and *description* is the text acquired by the one-hop relations of the entity v. Figure 4 shows our candidate entities ranking model, where Bert, Word2vec, Glove and KgCLUE baseline model are used as the baseline models for mention and entity encoding. To demonstrate the effect of our approach, we represented mentions and entities with the existing PLMs. At last, we ranked the candidate entities by calculating the cosine correlation between mention and candidate entities.

Negative sampling approach

We regarded a correctly corresponding mention–entity pair as a positive sample and an incorrectly corresponding pair as a negative sample. Semantic matching can be regard as a binary classification task that is assigned while training. We considered that a better model could be trained if we created a set of high-quality negative samples of mention–entity pairs. The conventional approach to creating negative samples that uses random or normal

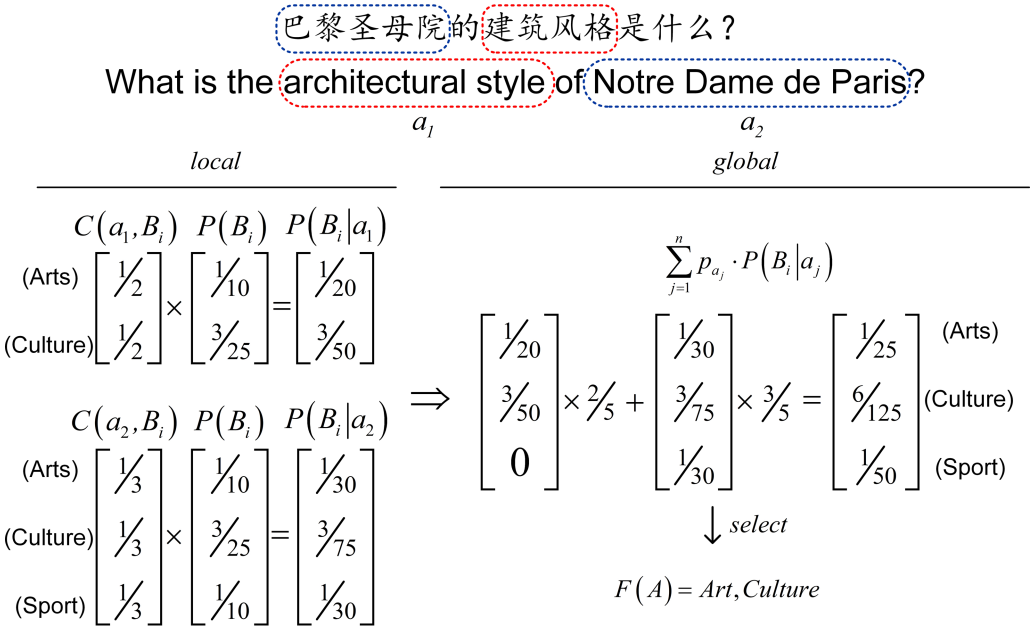

Figure 3 **An example of our model.**
Full-size DOI: 10.7717/peerjcs.1233/fig-3

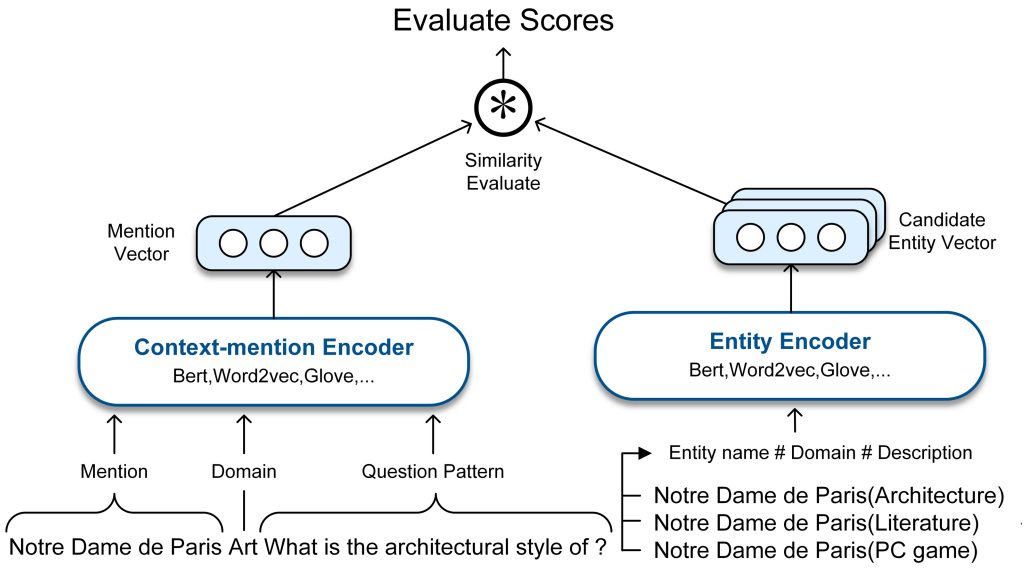

Figure 4 **The architecture of the candidate entities ranking model.**
Full-size DOI: 10.7717/peerjcs.1233/fig-4

distributions is blind and may even be deleterious in model training because it neglects any information that a sample takes (*Cai et al., 2020*).

An analysis of samples that were misclassified by the baseline models showed two typical situations: some pairs that are approximately semantically identical, such as "NBA" and

"NCAA", were identified as identical; the another is the surface form similar "University of York" (in the United Kingdom) and "York university" (in Toronto, Canada). Most approaches, for this situation, prefer negative sampling to enrich the training datasets and guide the learning process. Inspired by the representational approaches (*Rao, He & Lin, 2016*; *Zhang et al., 2019*), we considered both semantics and surface form for improve the quality of negative sampling.

For selecting the alternative options for our negative sampling strategy, we enumerated several traditional and popular approaches for measurement of proximity: Minkowski distance, Edit distance and Cosine distance. Minkowski distance and special cases based on it, such as Euclidean distance, Hamming distance and Chebyshev distance, are not friendly to high-dimensional vectors due to the low interpretability of physical meaning (*Wang & Dong, 2020*). Edit distance focuses on the morphological differences between the strings, which is sensitive to the surface dissimilarity. Cosine distance concentrates more on the difference of direction of vectors rather than absolute values. Compared with Minkowski distance, the Cosine distance emerge a better performance in proximity measurement of high-dimensional vectors. We thus prefer utilizing the Edit distance and Cosine distance in our negative sampling strategy.

For semantic matching, we used the pre-trained BERT model to encode the entities with descriptions and measured proximity by cosine distance:

$$similarity = cos(\theta) = \frac{A \cdot B}{\|A\| \cdot \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}}. \tag{8}$$

For surface form matching, we used Levenshtein distance ratio, a classical algorithm for Edit distance calculation:

$$r = (sum - ldist)/sum \tag{9}$$

where sum is the overall length of $str1$ and $str2$, and $ldist$ is the edit distance.

Intuitively, a good negative sample will have a balance between obviously similar and dissimilar items. We finally combine surface form and semantics in a rule for negative sample selection:

$$\begin{cases} 1, if\ \alpha < Lev_{\overline{e},e_i} < \beta\ and\ \gamma < Cos\ Sim(\overline{e}, e_i) < \delta \\ 0, \qquad otherwise \end{cases} \tag{10}$$

where $Lev_{\overline{e},e_i}$ and $CosSim(\overline{e}, e_i)$ are respectively the surface form and semantic similarities between $\overline{e}$ in positive mention-entity pairs and entities e_i in KB. The parameters $\alpha, \beta, \gamma, \delta$ are hyper parameters for tuning; the optimal values were respectively 0.9, 0.5, 0.8 and 0.6.

EXPERIMENT

Data preparation

CLUE is one of the most authoritative benchmarks in the field of Chinese language understanding. KgCLUE (*Xu et al., 2020*) is a carefully designed Chinese KBQA benchmark based on CLUE and combining the characteristics and recent development trends of KBQA,

which contains a KB, a QA dataset and several baseline model for different tasks. We chose the dataset provided by the large Chinese open source KgCLUE as the basic data source for our experiment; it contains 18,000 training pairs, 2,000 valid pairs, 2,000 test pairs and an original KB. To test the adaptability of our method we selected a dataset similar but owns more noise than KgCLUE, named NLPCC2016, which contains 14,609 training pairs and 9,870 test pairs. We chose KBQA datasets rather than entity linking datasets for two reasons: our approach to entity linking was directed towards KBQA; and most entity linking datasets are created from anchor text that targets web pages, which may be unrealistic for practical use.

Baseline models

In order to test whether our approach can improve the ability of entity linking models without structurally changing the models, we chose three widely used PLMs for mention and entity encoding: BERT (*Devlin et al., 2019*), Word2vec (*Mikolov et al., 2013*) and Glove (*Pennington, Socher & Manning, 2014*). We also experiment with KgCLUE provided baseline model for similarity calculation that was based on RoBERTa. In the experiment, we fine-tuned BERT and the KgCLUE baseline model and use the original Word2vec and Glove algorithm to generate variety representations.

Semantic matching results

We regard semantic matching as a binary classification task and test the effectiveness of our approach *via* it. To verify that the DME model processed data can be applied to multiple models, the BERT, KgCLUE provided baseline model, Word2vec and Glove are chosen as the baseline model. Table 3 shows the Accuracy and the F1 scores of three PLMs and KgCLUE provided baseline model with KgCLUE datasets. Fine-tuned BERT performed better than pretrained Word2vec and Glove. Compared with KgCLUE baseline model, fine-tuned BERT with DME-processed data increased by about 7% in accuracy and F1 scores. Obviously, the performance of these models has been improved to varying degrees with DME-processed data.

In addition, the KgCLUE and NLPCC2016 were chosen as the test datasets for adaptability of DME. Table 4 shows that the DME is suitable for different datasets and improved the accuracy and F1 score obviously.

After analyzing the results, we conjectured that DME explicitly enriches the feature dimension of the data in a manner that increases the diversity of mentions and entity representation. To verify this conjecture, we randomly sampled 1,000 positive pairs and 1,000 negative pairs for an experiment and designed a diversity score F_t to quantify the diversity:

$$F_t = \begin{cases} S_{with\ \text{domain information}} - S_{without\ \text{domain information}}, & positive\ samples \\ S_{without\ \text{domain information}} - S_{with\ \text{domain information}}, & negative\ samples \end{cases} \tag{11}$$

where S is the cosine similarity with or without supplementary domain information. Figures 5 and 6 show the results of the experiment, which verified our conjecture.

Table 3 Performance of all the models using the KgCLUE dataset with and without domain information as indicated by accuracy and F1 scores.

Models	Accuracy	F1 scores
Bert (Original datasets)	84.61%	84.14%
Bert (DME-processed datasets)	94.03%	95.23%
KgCLUE baseline model	87.20%	88.52%
Glove (Original datasets)	62.08%	65.26%
Glove (DME-processed datasets)	66.40%	67.35%
Word2Vec (Original datasets)	65.84%	65.96%
Word2Vec (DME-processed datasets)	70.71%	70.57%

Table 4 Performance of BERT using the KgCLUE and NLPCC2016 with and without DME treating as indicated by accuracy and F1 score.

	KgCLUE		NLPCC2016	
	Accuracy	F1 score	Accuracy	F1 score
BERT with original dataset	84.61%	85.28%	84.76%	84.07%
BERT with DME-processed datasets	94.03%	95.23%	86.35%	86.50%

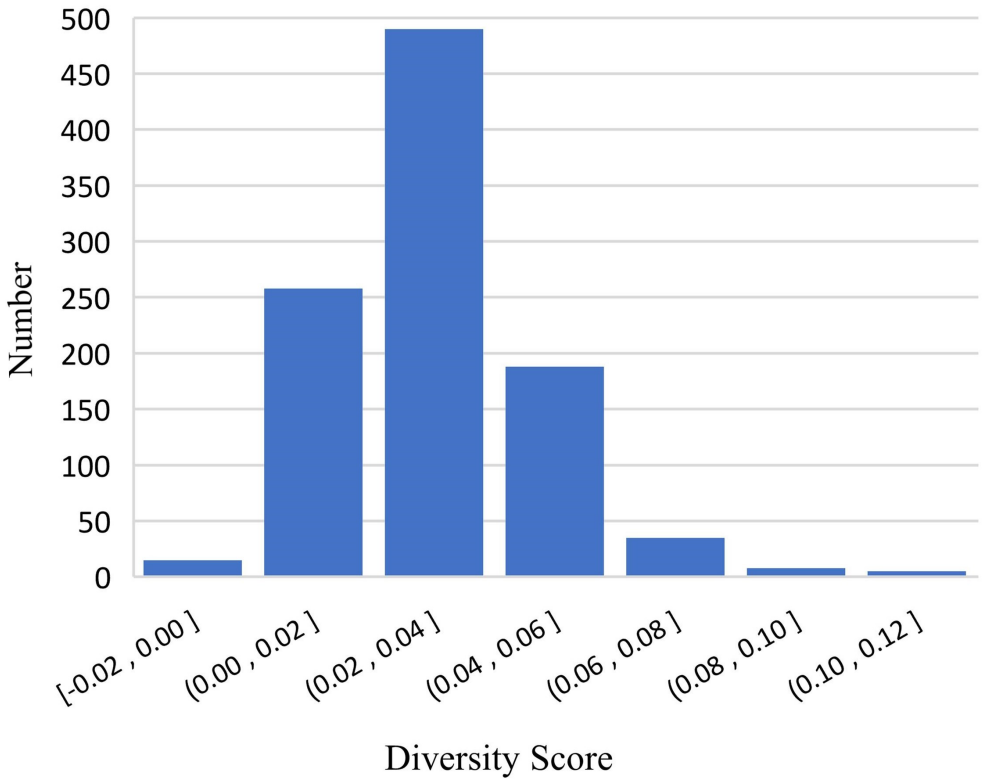

Figure 5 Differences in positive samples before and after DME processing.
Full-size DOI: 10.7717/peerjcs.1233/fig-5

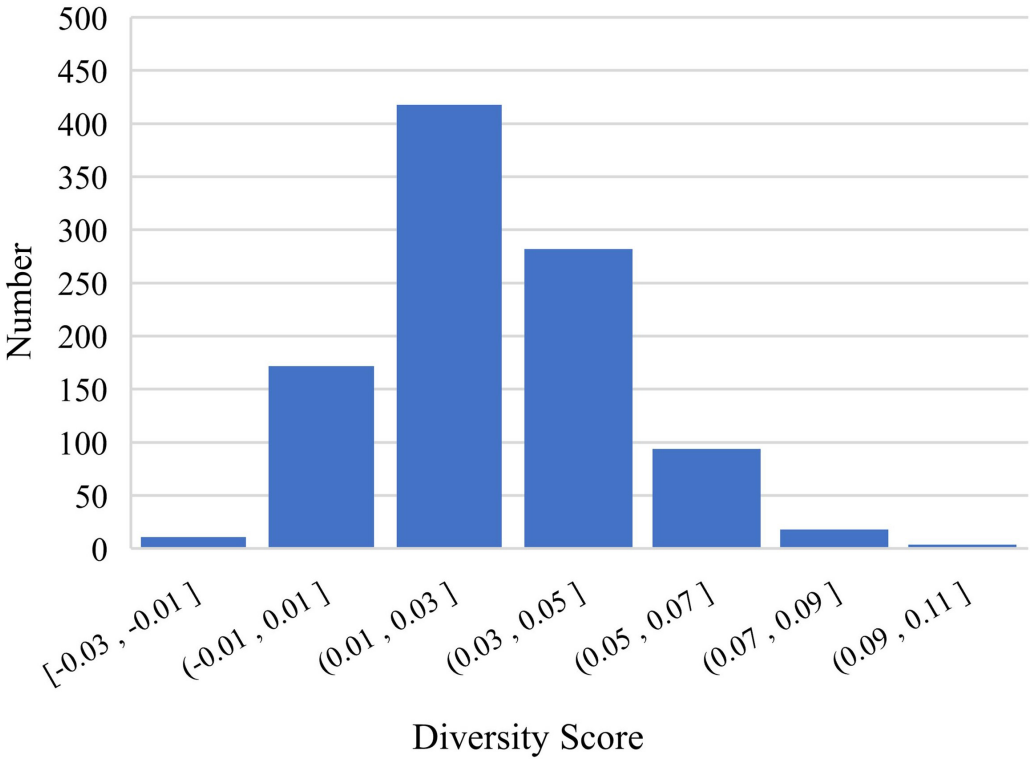

Figure 6 **Differences in negative samples before and after DME processing.**
Full-size DOI: 10.7717/peerjcs.1233/fig-6

Table 5 **Test classification accuracies with different negative sampling strategies.**

Method	Accuracy	F1	Time cost
Without negative sampling	84.61%	84.14%	–
Random	86.33%	85.27%	294′
Entity replacement with wordform and semantic	94.03%	95.23%	186′

Negative sampling results

To verify the effectiveness of our negative sampling strategy, with the DME-processed KgCLUE datasets, we compared our strategy with the random negative sampling strategy. Table 5 shows that our strategy produced a better result, which shows that hard negative samples better trained the model in terms of extending the decision boundary. In time overhead, our proposed method reduces 36.73% compared to random sampling. Table 6 shows the accuracy for different numbers of negative samples, one positive sample with two negative samples gets the highest measurement scores.

Ablation experiment results

To verify each individual component that we put forward to improve the entity linking task, an ablation experiment was proposed. In our research, DME, random negative sampling strategies and negative sampling strategies combining wordform and semantic are all approaches that can affect the quality of the dataset and, furthermore, the performance of

Table 6 Test classification accuracies with different number of negative samples.

Num	1	2	3	4	6	8	10	12
Acc	92.36%	94.01%	93.85%	93.67%	93.76%	93.79%	93.76%	93.02%
Recall	89.27%	91.64%	91.08%	90.53%	90.65%	90.80%	90.92%	89.55%
F1	93.14%	94.95%	94.81%	94.71%	94.80%	94.79%	94.76%	94.20%

Table 7 The impact of each approach and their combination on the entity linking task with the KgCLUE dataset.

BERT	DME	Random	Semantic	Wordform	Accuracy	F1
√					84.61%	84.14%
√	√				88.90%	91.83%
√		√			81.68%	82.93%
√			√		90.32%	88.80%
√				√	88.34%	87.75%
√			√	√	91.90%	90.48%
√	√		√	√	94.03%	95.23%

the model. We therefore use these approaches to process the KgCLUE dataset separately or jointly and fine-tune a BERT model with different datasets. Table 7 shows the impact of each approach and their combination on the entity linking task.

Discussion

In this research, we increased the robustness of EL in the model without changing the model structure by designing a data-centric model, DME, to mine domain information for short text. Besides, an innovative negative sampling approach that considering both surface form and semantic information was proposed to generate high quality negative samples. Overall, the three main parts of experiments support the methods well. In this subsection we will have a discussion with the results above.

We implemented a set of experiments to test the effectiveness and transferable ability of DME. Table 3 shows the Accuracy and the F1 scores of three PLMs and KgCLUE provided baseline model with KgCLUE datasets. Obviously, the performance of these models has been improved to varying degrees with DME-processed data. The results supported our intuition that DME-processed data can improve model performance without necessitating structural change to the original models. Table 4 shows that the DME is suitable for different datasets. It improved the accuracy and F1 score clearly. After analyzing the results, we conjectured that DME explicitly enriches the feature dimension of the data in a manner that increases the diversity of mentions and entity representation. Figures 5 and 6 verified the conjecture. Influenced by the errors when DME recognizing the domain of a mention or an entity, some of the F_t are negative numbers, which will be improved by optimizing the dictionaries.

The limited datasets and models are bounded in testing and verifying the transferable ability of our approach. The core concept of DME is to improve the performance of the model by enhancing the data quality. In fact, we usually preprocess data when training

models, which is consistent with the concept of "data centric". The reason is that we believe that better datasets can advance model training. So, we believe that DME has outstanding portability in theory, that is, it can be applied to most datasets. In the meantime, it can be effective for stronger models than BERT.

As to the negative sampling strategy that we proposed, there are two experiments raised to verify the effectiveness. Table 5 shows that our strategy produced a better result, which shows that hard negative samples better trained the model in terms of extending the decision boundary. As to the time cost, our method is much lower than random sampling. This is because a good negative sampling strategy is able to reduce the time complexity of gradient descent when fine-tuning the BERT. Sampling randomly may introduce wrongly labeled mention-entity pairs unpredictability that caused the limited improvement and higher time overhead. The results of the experiment also verified the conclusion that we proposed in the 'Negative sampling approach' section. Table 6 shows the accuracy for different numbers of negative samples, we inferred that one positive sample with two negative samples can optimally train the model.

In the ablation experiment part, Table 7 shows the impact of each approach and their combination on the entity linking task. By analyzing the experimental results, we may draw the following conclusions: (1) Our proposed DME model can effectively improve the performance of entity linking tasks; (2) The random sampling process will generate some false negative samples, which will make adverse impact on the model training; (3) In contrast, considering semantic information is more conducive to high-quality negative sampling than considering wordform information; (4) The combination of our proposed DME model and negative sampling strategy can achieve better results on entity linking task.

CONCLUSIONS

In this article, we built a domain dictionary and then proposed a lightweight and effective data-centric model, DME, to mine and explicitly express domain information relating to mention and entity. In addition, we proposed a negative sampling strategy that considering both semantic and wordform information of the text. According to the experimental observation, the quantity and quality of negative samples can affect the performance of an entity linking model. Also, the combination of our proposed DME model and negative sampling strategy can achieve better results on entity linking task. The future work will be concentrated on three directions: First, improving the quality of domain dictionaries by expert inspection to increase the accuracy of DME; second, proving the effectiveness of our proposed method on some recently proposed advanced models, third, extending the DME to more tasks such as the text classification and search engine.

ADDITIONAL INFORMATION AND DECLARATIONS

Funding

This research is supported by the Science and Technology Research Program of the Department of Science and Technology of Henan Province (approval No.: 222102210081). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Grant Disclosures

The following grant information was disclosed by the authors:
Science and Technology Research Program of the Department of Science and Technology of Henan Province: 222102210081.

Competing Interests

The authors declare there are no competing interests.

Author Contributions

- Shuo Liu conceived and designed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, and approved the final draft.
- Gang Zhou conceived and designed the experiments, analyzed the data, authored or reviewed drafts of the article, and approved the final draft.
- Yi Xia conceived and designed the experiments, analyzed the data, prepared figures and/or tables, and approved the final draft.
- Hao Wu performed the experiments, performed the computation work, authored or reviewed drafts of the article, and approved the final draft.
- Zhufeng Li performed the experiments, performed the computation work, prepared figures and/or tables, and approved the final draft.

Data Availability

The following information was supplied regarding data availability:
The code and data are available in the Supplemental Files.

Supplemental Information

Supplemental information for this article can be found online at http://dx.doi.org/10.7717/peerj-cs.1233#supplemental-information.

REFERENCES

Auer S, Bizer C, Kobilarov G, Lehmann J, Cyganiak R, Ives Z. 2007. Dbpedia: a nucleus for a web of open data. In: *The semantic web. ISWC ASWC 2007 2007. Lecture Notes in Computer Science*, vol. 4825. Berlin, Heidelberg: Springer, 722–725 DOI 10.1007/978-3-540-76298-0_52.

Bollacker K, Evans C, Paritosh P, Sturge T, Taylor J. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In: *Proceedings of the 2008*

ACM SIGMOD international conference on management of data. New York: ACM, 1247–1250 DOI 10.1145/1376616.1376746.

Cai TT, Frankle J, Schwab DJ, Morcos AS. 2020. Are all negatives created equal in contrastive instance discrimination? ArXiv preprint. arXiv:abs2010.06682.

Chen S, Wang J, Jiang F, Lin C-Y. 2020. Improving entity linking by modeling latent entity type information. In: *The thirty-fourth AAAI conference on artificial intelligence, AAAI 2020, The thirty-second innovative applications of artificial intelligence conference, IAAI 2020, The tenth AAAI symposium on educational advances in artificial intelligence, EAAI 2020, New York, NY, USA, February (2020) 7-12*. Palo Alto: AAAI, 7529–7537.

Devlin J, Chang M-W, Lee K, Toutanova K. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In: Burstein J, Doran C, Solorio T, eds. *Proceedings of the 2019 conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*. Cedarville: ACL, 4171–4186 DOI 10.18653/v1/n19-1423.

Eshel Y, Cohen N, Radinsky K, Markovitch S, Yamada I, Levy O. 2017. Named entity disambiguation for noisy text. In: Levy R, Specia L, eds. *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017), Vancouver, Canada, August (2017) 3-4*. Cedarville: ACL, 58–68 DOI 10.18653/v1/K17-1008.

Fang Z, Cao Y, Zhang D, Li Q, Zhang Z, Liu Y. 2019. Joint entity linking with deep reinforcement learning. ArXiv preprint. arXiv:abs1902.00330.

Francis-Landau M, Durrett G, Klein D. 2016. Capturing semantic similarity for entity linking with convolutional neural networks. In: Knight K, Nenkova A, Rambow O, eds. *NAACL HLT 2016, The 2016 conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June (2016), 12-17*. Cedarville: ACL, 1256–1261 DOI 10.18653/v1/n16-1150.

Huang X, Zhang J, Xu Z, Ou L, Tong J. 2021. A knowledge graph based question answering method for medical domain. *PeerJ Computer Science* **7**:e667 DOI 10.7717/peerj-cs.667.

Lample G, Ballesteros M, Subramanian S, Kawakami K, Dyer C. 2016. Neural architectures for named entity recognition. In: Knight K, Nenkova A, Rambow O, eds. *NAACL HLT 2016, The 2016 conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June (2016), 12-17*, 260–270 DOI 10.18653/v1/n16-1030.

Le P, Titov I. 2019. Distant learning for entity linking with automatic noise detection. In: Korhonen A, Traum DR, Màrquez L, eds. *Proceedings of the 57th conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28-August 2, 2019, Volume 1: Long Papers*. Cedarville: ACL, 4081–4090 DOI 10.18653/v1/p19-1400.

Logeswaran L, Chang M-W, Lee K, Toutanova K, Devlin J, Lee H. 2019. Zero-shot entity linking by reading entity descriptions. ArXiv preprint. arXiv:abs1906.07348.

Lu X, Deng Y, Sun T, Gao Y, Feng J, Sun X, Sutcliffe R. 2022. MKPM: multi keyword-pair matching for natural language sentences. *Applied Intelligence* **52**:1878–1892 DOI 10.1007/s10489-021-02306-5.

McIlwaine IC. 1997. The universal decimal classification: some factors concerning its origins, development, and influence. *Journal of the American Society for Information Science* **48**:331–339 DOI 10.1002/(SICI)1097-4571(199704)48:4<331::AID-ASI6>3.0.CO;2-X.

Mikolov T, Chen K, Corrado G, Dean J. 2013. Efficient estimation of word representations in vector space. In: Bengio Y, LeCun Y eds. *1st international conference on learning representations, ICLR 2013, Scottsdale, Arizona, USA, May (2013) 2-4, workshop track proceedings*.

Moreno JG, Besançon R, Beaumont R, D'hondt E, Ligozat A-L, Rosset S, Tannier X, Grau B. 2017. Apprendre des représentations jointes de mots et d'entités pour la désambiguïsation d'entités (Combining Word and Entity Embeddings for Entity Linking). In: Eshkol-Taravella I, Antoine J-Y, eds. *Actes des 24ème conférence sur le traitement automatique des Langues Naturelles, TALN 2017, Orléans, France, June (2017), 26-30, Volume 1, Articles longs*. Cedarville: ACL, 182–195.

Mulang' IO, Singh K, Prabhu C, Nadgeri A, Hoffart J, Lehmann J. 2020. Evaluating the impact of knowledge graph context on entity disambiguation models. In: d'Aquin M, Dietze S, Hauff C, Curry E, Cudré-Mauroux P, eds. *CIKM '20: The 29th ACM international conference on information and knowledge management, virtual event, Ireland, October (2020), 19-23*. New York: ACM, 2157–2160 DOI 10.1145/3340531.3412159.

Ng A. 2021. A chat with Andrew on MLOps: from model-centric to data-centric AI.

Nie F, Cao Y, Wang J, Lin C-Y, Pan R. 2018. Mention and entity description co-attention for entity disambiguation. In: McIlraith SA, Weinberger KQ, eds. *Proceedings of the thirty-second AAAI conference on artificial intelligence, (AAAI-18), the 30th innovative applications of artificial intelligence (IAAI-18), and the 8th AAAI symposium on educational advances in artificial intelligence (EAAI-18), New Orleans, Louisiana, USA, February (2018) 2-7*. Palo Alto: AAAI, 5908–5915.

Pennington J, Socher R, Manning CD. 2014. Glove: global vectors for word representation. In: Moschitti A, Pang B, Daelemans W, eds. *Proceedings of the 2014 conference on empirical methods in natural language processing, EMNLP 2014, October (2014), 25-29, Doha, Qatar, a meeting of SIGDAT, a special interest group of the ACL*. Cedarville: ACL, 1532–1543 DOI 10.3115/v1/d14-1162.

Peters ME, Neumann M, RLL IV, Schwartz R, Joshi V, Singh S, Smith NA. 2019. Knowledge Enhanced Contextual Word Representations. ArXiv preprint. arXiv:abs1909.04164.

Rao J, He H, Lin J. 2016. Noise-contrastive estimation for answer selection with deep neural networks. In: Mukhopadhyay S, Zhai C, Bertino E, Crestani F, Mostafa J, Tang J, Si L, Zhou X, Chang Y, Li Y, Sondhi P, eds. *Proceedings of the 25th ACM international conference on information and knowledge management, CIKM 2016, Indianapolis, IN, USA, October (2016), 24-28*. New York: ACM, 1913–1916 DOI 10.1145/2983323.2983872.

Schindler D, Bensmann F, Dietze S, Krüger F. 2022. The role of software in science: a knowledge graph-based analysis of software mentions in PubMed Central. *PeerJ Computer Science* **8**:e835 DOI 10.7717/peerj-cs.835.

Sevgili Ö, Shelmanov A, Arkhipov MY, Panchenko A, Biemann C. 2022. Neural entity linking: a survey of models based on deep learning. *Semantic Web* **13**:527–570 DOI 10.3233/SW-222986.

Shen W, Wang J, Han J. 2015. Entity linking with a knowledge base: issues, techniques, and solutions. *IEEE Transactions on Knowledge and Data Engineering* **27**:443–460 DOI 10.1109/TKDE.2014.2327028.

Sil A, Kundu G, Florian R, Hamza W. 2018. Neural cross-lingual entity linking. In: McIlraith SA, Weinberger KQ, eds. *Proceedings of the thirty-second AAAI conference on artificial intelligence, (AAAI-18), the 30th innovative applications of artificial intelligence (IAAI-18), and the 8th AAAI symposium on educational advances in artificial intelligence (EAAI-18), New Orleans, Louisiana, USA, February (2018) 2-7*. Palo Alto: AAAI, 5464–5472.

Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I. 2017. Attention is all you need. In: Guyon I, von LuxburgU, Bengio S, Wallach HM, Fergus R, Vishwanathan SVN, Garnett R. eds. *Advances in neural information processing systems 30: annual conference on neural information processing systems 2017, December (2017) 4–9, Long Beach, CA, USA*. 5998–6008.

Wang J, Dong Y. 2020. Measurement of text similarity: a survey. *Information* **11**:421 DOI 10.3390/info11090421.

Wu L, Petroni F, Josifoski M, Riedel S, Zettlemoyer L. 2020. Scalable zero-shot entity linking with dense entity retrieval. In: Webber B, Cohn T, He Y, Liu Y, eds. *Proceedings of the 2020 conference on empirical methods in natural language processing, EMNLP 2020, Online, November (2020), 16-20*. Cedarville: ACL, 6397–6407 DOI 10.18653/v1/2020.emnlp-main.519.

Xu L, Hu H, Zhang X, Li L, Cao C, Li Y, Xu Y, Sun K, Yu D, Yu C, Tian Y, Dong Q, Liu W, Shi B, Cui Y, Li J, Zeng J, Wang R, Xie W, Li Y, Patterson Y, Tian Z, Zhang Y, Zhou H, Liu S, Zhao Z, Zhao Q, Yue C, Zhang X, Yang Z, Richardson K, Lan Z. 2020. CLUE: a Chinese language understanding evaluation benchmark. In: Scott D, Bel N, Zong C, eds. *Proceedings of the 28th international conference on computational linguistics, COLING 2020, Barcelona, Spain (Online), December (2020) 8–13*, Cedarville: ACL, 4762–4772 DOI 10.18653/v1/2020.coling-main.419.

Yamada I, Washio K, Shindo H, Matsumoto Y. 2019. Global entity disambiguation with pretrained contextualized embeddings of words and entities. ArXiv preprint. arXiv:1909.00426.

Zhang Y, Yao Q, Shao Y, Chen L. 2019. NSCaching: simple and efficient negative sampling for knowledge graph embedding. In: *35th IEEE international conference on data engineering, ICDE 2019, Macao, China, April (2019) 8-11*. Piscataway: IEEE, 614–625 DOI 10.1109/ICDE.2019.00061.

Zwicklbauer S, Seifert C, Granitzer M. 2016. Robust and collective entity disambiguation through semantic embeddings. In: Perego R, Sebastiani F, Aslam JA, Ruthven I,

Zobel J, eds. *Proceedings of the 39th international ACM SIGIR conference on research and development in information retrieval, SIGIR 2016, Pisa, Italy, July (2016), 17-21*. New York: ACM, 425–434 DOI 10.1145/2911451.2911535.