International Journal of Information Technology and Web Engineering
Volume 17 • Issue 1

Extracting Entity Synonymous Relations via Context-Aware Permutation Invariance

Nan Yan, Anhui Polytechnic University, China
Subin Huang, Anhui Polytechnic University, China
Chao Kong, Anhui Polytechnic University, China

ABSTRACT

Discovering entity synonymous relations is an important work for many entity-based applications. Existing entity synonymous relation extraction approaches are mainly based on lexical patterns or distributional corpus-level statistics, ignoring the context semantics between entities. For example, the contexts around ''apple'' determine whether ''apple'' is a kind of fruit or Apple Inc. In this paper, an entity synonymous relation extraction approach is proposed using context-aware permutation invariance. Specifically, a triplet network is used to obtain the permutation invariance between the entities to learn whether two given entities possess synonymous relation. To track more synonymous features, the relational context semantics and entity representations are integrated into the triplet network, which can improve the performance of extracting entity synonymous relations. The proposed approach is implemented on three real-world datasets. Experimental results demonstrate that the approach performs better than the other compared approaches on entity synonymous relation extraction task.

KEYWORDS

Context Semantic, Entity Synonymous Relation, Permutation Invariance, Triplet Network

INTRODUCTION

An entity synonymous relation is a semantic relationship between a pair of terms representing the same entity in the real world with the same or similar meaning (Abu-Salih, 2021; Qu et al., 2017; Shen et al., 2019). For example, $\left(\text{United States} \xrightarrow{Syn} \text{USA}\right)$ is a pair of entity synonymous relation, since the "United States" and the "'USA" both represent the same country: The "United States of America." In the specific applications, entity synonymous relations play an important role in many entity-based tasks, such as taxonomy construction (Abu-Salih et al., 2018; Huang et al., 2019; Huang et al., 2020; Wang et al., 2019), document retrieval (Kong et al., 2019; Liu et al., 2016; Wongthongtham et al., 2018; Yin et al., 2016), and topic detection (Padmanabhanet al., 2017; Xie et al., 2015). Therefore, extracting entity synonymous relations automatically is a crucial work for many downstream applications.

In previous work, the entity synonymous relation extraction approaches are mainly using lexical patterns or distributional corpus-level statistics:

DOI: 10.4018/IJITWE.288039

- **Lexical Pattern-Based Approaches:** Such approaches employ lexical patterns to mine entity synonymous relations from texts (Nguyen et al., 2017; Simanovsky et al., 2011; Wang et al., 2010). For example, given a lexical pattern "X is referred to Y" and a sentence "The acetylsalicylic acid is often referred to as the aspirin," it is possible to use the above lexical pattern to infer that "acetylsalicylic acid" and "aspirin" are synonymous.
- **Distribution-Based Approaches:** Such approaches exploit distributional corpus-level statistics to mine entity synonymous relations from texts (Chakrabarti et al., 2012; Qu et al., 2017; Turney, 2001). Based on the distributional hypothesis (Harris, 1954), the distribution-based approaches hold that terms that often appear in similar or same contexts are likely to be synonymous (Qu et al., 2017).

However, the above approaches have the following limitations:

- **Low Coverage and Weak Ability in Processing Complex Texts:** Lexical pattern-based approaches use the lexical patterns to mine the entity synonymous relations and thus result in low coverage. This is because it is difficult for the lexical patterns to effectively obtain entity synonymous relations from complex text.
- **Low Precision and Wrong Entity Synonymous Relation Label:** Distribution-based approaches may bring some noise. Some nonsynonymous entities can also appear in similar or same contexts. For example, "UK" and "USA" often appear in similar contexts, which could be labeled as a wrong entity synonymous relation pair.
- **Little Attention Paid on Context Semantics:** Lexical pattern-based and distribution-based approaches pay less attention on context semantics, and thus it is difficult to balance precision and recall.

In order to address the above limitations, this paper proposes an entity synonymous relation extraction approach based on context-aware permutation invariance. Specifically, the triplet network is employed to learn the permutation invariance (Huang et al., 2020; Shen et al., 2019) between the entities, and the entity relational contexts are employed to enhance the synonymous training signals in the triplet network. The main contribution of the paper is as follows:

- An improved triplet network framework is proposed to capture the permutation invariance between the entities, which is capable of learning whether two given entities possess synonymous relation.
- The relational context semantics among entities are integrated into the triplet network framework. The authors not only use the representations of entities, but also the relational context semantics among entities to capture the synonymous training signals, which is capable of improving the performance of the triplet network framework in mining entity synonymous relations.
- The authors conduct our proposed on three real-world datasets. Experimental results demonstrate the effectiveness of their approach, which outperforms the other compared approaches on entity synonymous relation extraction task.

The rest of the paper is structured as follows: The second section introduces the related work; the third section presents some terminologies and a basic idea of the paper; the fourth section details an implementation process of the authors' approach; the fifth section reports the experimental results; finally, the sixth section gives a conclusion of the paper.

RELATED WORK

Pattern-Based Approaches

In the early studies, researchers used manually designed textual patterns to extract synonyms from corpus (Roller et al., 2018; Wang et al., 2010). For example, from the sentence "X is referred to Y," where X and Y are nouns or noun phrases, it is possible to infer that X and Y have a synonymous relation. However, it is time-consuming and laborious to manually design synonymous textual patterns. Therefore, the subsequent studies mainly focus on how to automatically construct synonymous textual patterns from the corpus.

Wang et al. (2010) proposed a method to automatically extract synonymous and antonymous patterns from the corpus. They combined multiple different patterns to improve the coverage of extracting synonyms and antonyms. Simanovsky et al. (2011) first extracted the seed synonyms from Wikipedia and employed these seed synonyms to automatically generate synonymous patterns. Then, they evaluated the confidence of each synonymous pattern and exploited the high confidence synonymous patterns to extract novel synonyms from the corpus. In order to capture more synonymous and antonymous pattern features, Nguyen et al. (2017) integrated pattern paths and neural networks to extract synonyms and antonyms.

Distribution-Based Approaches

The distribution-based approaches employ corpus-level statistics information to extract synonyms from the corpus. Such approaches assume that terms appearing in the same or similar context are more likely to have approaches relation (Qu et al., 2017).

In order to capture more synonymous features from textual patterns and distributional statistics, Qu et al. (2017) employed distance supervision to extract seed synonyms from the knowledge base and integrate distributional statistics and textual patterns to automatically discover synonyms from the corpus. Turney (2001) proposed an unsupervised synonym recognition method based on pointwise mutual information (PMI) and information retrieval (IR). In this method, the candidate words with the most similar meaning to the target words are selected as synonyms of the target words by using PMI-IR algorithm. Chakrabarti et al. (2012) used two similar functions for building an extraction framework of extracting synonyms, and applied MapReduce to this framework to extract synonyms efficiently, scalable, and large-scale.

Encyclopedia-Based Approaches

Encyclopedia resources (e.g., infoboxes) are good data sources for semantic relation extraction (Sottovia et al., 2019). Some researchers have investigated the automatic synonym extraction from the encyclopedia by using textual patterns and statistics information.

Sottovia et al. (2019) proposed a method to acquire synonymous attributes from Wikipedia infoboxes. They employed cooccurrence degree and similar values to identify clusters of synonymous infobox attributes. Lu and Hou (2008) proposed an automatic Chinese synonym extraction method based on Wikipedia. First, they designed some acquiring patterns to extract synonyms from the definition content. Second, they employed the PageRank algorithm to generate synonyms from the associated word graph. Jagannatha et al. (2015) extracted and ranked candidate biomedical synonyms from inter-wiki links of Wikipedia. They employed distributed word representation and pseudo relevance feedback to improve the quality of synonyms.

Discussion

In the previous subsections, the authors reviewed three entity synonym extraction approaches. The pattern-based approaches can achieve high precision; however, such approaches suffer from low coverage. The distribution-based approaches often suffer from low precision because some nonsynonymous terms (e.g., "banana" and "apple") may have similar contexts. The encyclopedia-based approaches usually extract entity synonym relations from a single resource (e.g., infoboxes or inter links), which may result in low coverage.

In this paper, the authors focus on the problem of extracting entity synonymous relations via context-aware permutation invariance. The researchers employ the triplet network to capture the permutation invariance between the entities, which is capable of learning whether two given entities possess synonymous relation. Moreover, to track more synonymous training signals, they integrate the relational context semantics and entity representations into the triplet network, which is capable of improving the performance of our approach on the entity synonymous relation extraction task.

Terminologies and Problem Definition

This section gives several terminologies and a problem definition for the proposed approach.

- **Synonymous Relation:** A synonymous relation is a semantic relationship between a pair of terms with the same or similar meaning in the real-world (Shen et al., 2019). For example, "willpower" and "strength of will" are synonymous, which refer to a very strong determination to do something; "United States" and "USA" are synonymous, which refer to the same country.
- **Entity Synonymous Relation:** An entity synonymous relation is a semantic relationship between a pair of terms representing the same entity in the real world (Qu et al., 2017). For example, "caustic soda" and "sodium hydroxide" are synonymous because the "caustic soda" is the alternative name of the inorganic compound entity "sodium hydroxide;" "aspirin" and "acetylsalicylic acid" are synonymous because the "aspirin" is the alternative name of the drug entity "acetylsalicylic acid."
- **Permutation Invariance of Synonymous Entities:** Based on the definition of entity synonymous relation, the permutation invariance of synonymous entities is obvious (Shen et al., 2019). Given two entities e_i and e_j, if e_i and e_j are synonymous, then e_i and e_j are capable of replacing each other in real-world language expression. For example, the synonymous entities "mom" and "mother" both can appear in the asterisk of a sentence "Her * seemed very amiable."
- **Problem Definition:** Based on the above terminologies, the problem of the paper is formally defined as follows: Given a plain text corpus C and an entity vocabulary E mined from C, the authors' problem focuses on extracting the entity synonymous relation pairs from E.

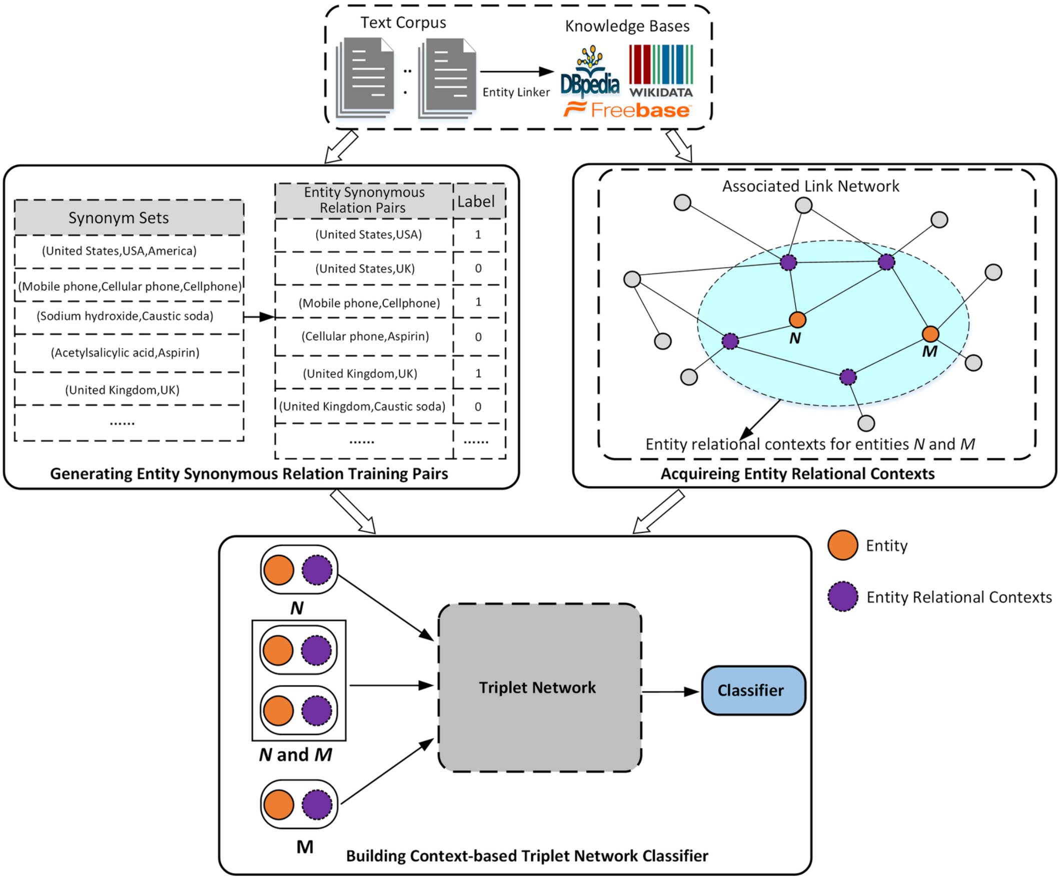
APPROACH

Figure 1 shows the overall framework of the authors' approach. First, they generate entity synonymous relation training pairs based on distant supervision and knowledge bases. Second, they present an entity relational contexts acquisition method. Third, the researchers integrate the acquired entity relational contexts into the triplet network classifier to learn whether two given entities possess synonymous relation.

Generating Entity Synonymous Relation Training Pairs

In order to generate training datasets automatically, Minize et al. (2009) proposed a distant supervision strategy to retrieve training datasets from existing knowledge bases and plain text corpus. The workflow of distant supervision consists of three steps (Qu et al., 2017). First, entity mentions are discovered in plain text corpora. Second, based on an existing knowledge base, the discovered entity mentions are mapped to the entities involved in the knowledge base. Third, the training datasets are collected from the knowledge base. In this paper, the authors acquire the entity synonymous relation training pairs from the distant supervision acquisition (e.g., entity synonym sets) collected by Qu et al. (2017) and Shen et al. (2019).

Figure 1. The overall framework



Given an entity synonym sets $EntSynSet = \left(S_1, \cdots, S_k, \cdots, S_N\right)$, for each entity synonym set $S_k = \left(e_1, \cdots, e_n\right)$ in $EntSynSet$, the authors first select two entities e_i and e_j from S_k and regard $\left(e_i \xrightarrow{Syn} e_j\right)$ as a positive entity synonymous relation training pair. Then, they select $2{*}m$ negative entities $e_k^{neg}\,|_1^{2m}$ from $EntSynSet = \left(S_1, \cdots, S_{k-1}, S_{k+1} \cdots, S_N\right)$, and construct $2{*}m$ negative entity synonymous relation training pairs: $\left(e_i \xrightarrow{Non-Syn} e_k^{neg}\,|_1^{m}\right)$ and $\left(e_k^{neg}\,|_{m+1}^{2m} \xrightarrow{Non-Syn} e_j\right)$.

Acquiring Entity Relational Contexts

In this section, the authors employ associated link network (ALN) (Luo et al., 2011) and relational paths (Wang et al., 2020) to acquire entity relational contexts. Specifically, they first build ALN from a plain text corpus. Then, they exploit the relational paths of the entities involved in ALN to acquire entity relational contexts.

ALN aims to establish the relationship between terms involved in a plain text corpus, which is mainly composed of key elements and semantic chains. The key elements are terms such as entities and keywords, and the semantic chains are used to express the strength of association between key elements. As Figure 2 shows, the ALN is built based on association rules and context similarity.

Step 1: Building basic ALN. The TF-IDF algorithm is employed to extract key elements from a text corpus. The association rules mining algorithm (Han et al., 2000) is used to construct basic ALN:

$$ALN = \left\{Terms, Links\right\} \tag{1}$$

$$Terms = \left(t_1, \cdots, t_N\right) \tag{2}$$

$$Links = \begin{pmatrix} l_{11} & \cdots & l_{1N} \\ \vdots & \ddots & \vdots \\ l_{N1} & \cdots & l_{NN} \end{pmatrix} \tag{3}$$

where $Terms$ is a set of key elements, $Links$ are the strength of semantic chains between semantic elements, and N is the number of semantic elements.

Step 2: Calculating semantic chain strength. The context similarity method is used to calculate the semantic chain strength between key elements:

$$l_{ij} = sim\left(t_i, t_j\right) = \frac{v_{t(i)} v_{t(j)}}{v_{t(i)} v_{t(j)}} \tag{4}$$

where $v_{t(i)}$ and $v_{t(j)}$ are the word representations of semantic elements t_i and t_j.

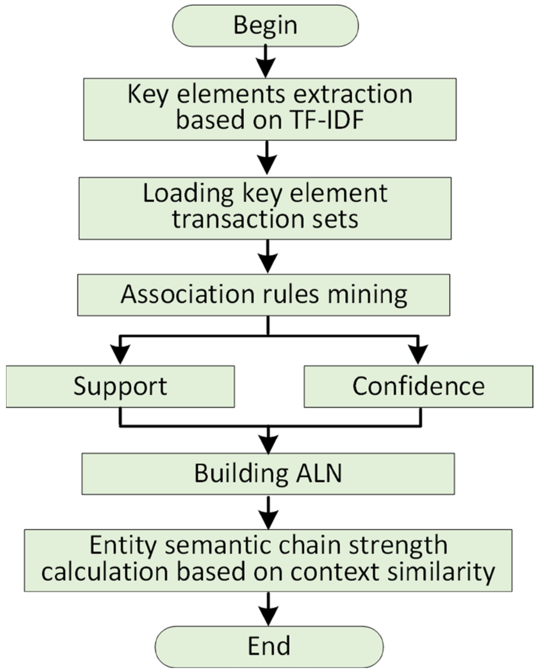
Based on the above ALN, the authors exploit the relational paths of the entities involved in ALN to acquire entity relational contexts. Given an entity synonymous relation training pair $\left(e_i \xrightarrow{Syn} e_j\right)$, they retrieve all one-hop and two-hop relational paths from e_i to e_j in ALN. Each entity involved in the one-hop, and two-hop relational paths is considered as the entity relational contexts of e_i and e_j. For example, as Figure 3 shows, the entity relational contexts of "United States" are "Government, President, Citizen," and the entity relational contexts of "USA" are "President, Green card."

Building Context-Aware Triplet Network Classifier

In this section, the authors present a context-aware triplet network classifier to discover entity synonymous relations. In the classifier, the triplet network is employed to learn the permutation invariance between the synonymous entities, while the entity relational contexts are exploited to enhance the synonymous training signals in the triplet network.

As Figure 4 illustrates, the context-aware triplet network classifier $f\left(N, M\right)$ uses not only the representations of entities, but also the relational context semantics among entities to capture the permutation invariance between synonymous entities.

Figure 2. The construction flow chart of ALN



Step 1: Given an entity pair (N, M), the entity relational contexts between N and M, and an embedding lookup table, the classifier $f(N, M)$ obtains e_N (an embedding of entity N), e_M (an embedding of entity M), e_{NC} (the embeddings of entity relational contexts for entity N), and e_{MC} (the embeddings of entity relational contexts for entity M).

Step 2: The authors employ a triplet network (triplet siamese network with five hidden layers) to capture the permutation invariance between synonymous entities. The inputs of the triplet network are R_{NC}, R_{NMC} and R_{MC}, where $R_{NC} = e_N \oplus e_{NC}$, $R_{NMC} = e_N \oplus e_{NC} + e_M \oplus e_{MC}$ and $R_{MC} = e_M \oplus e_{MC}$. The outputs of the triplet network are hidden representations H_{NC}, H_{NMC} and H_{MC}.

Step 3: The classifier $f(N, M)$ first computes the difference between H_{NMC} and H_{NC}, and the difference between H_{NMC} and H_{MC}. Then, the classifier $f(N, M)$ feeds these two differences into a sigmoid unit to get probability:

$$f(N, M) = sigmoid\left(\left(H_{NMC} - H_{NC}\right) + \left(H_{NMC} - H_{MC}\right)\right) \tag{5}$$

Given a dataset of entity synonymous relation training pairs $\left\{\left(u_i, v_i, y_i\right)|_{i=1}^{K}\right\}$, the researchers train the context-based triplet network classifier by using the log cross-entropy loss:

Figure 3. Examples of the relational paths between entities involved in ALN

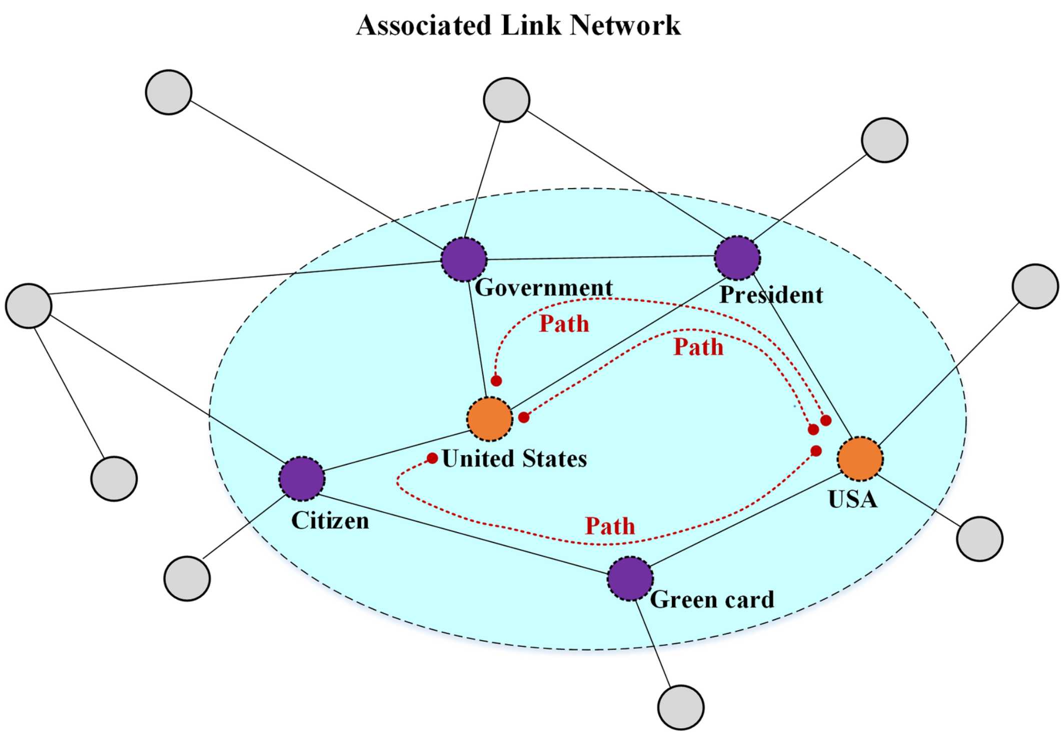
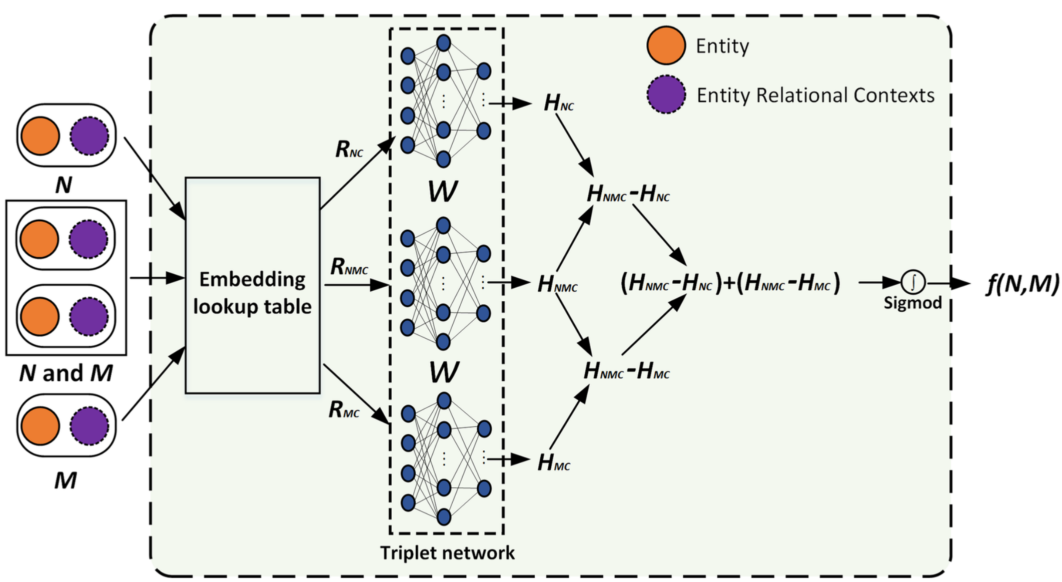


Figure 4. The architecture of the context-aware triplet network classifier



$$\mathcal{L} = -\frac{1}{K}\sum_{i=1}^{K}\left[log\left(f\left(u_i, v_i\right)\right) * y_i + log\left(1 - f\left(u_i, v_i\right)\right) * \left(1 - y_i\right)\right] \tag{6}$$

where K is the number of entity synonymous relation training pairs. y_i equals to 1 if u_i and v_i are synonymous and to 0 otherwise.

EXPERIMENT

Experimental Setup

Datasets

Based on the datasets released by Qu et al. (2017) and Shen et al. (2019), the authors generate three entity synonymous relation training datasets named NYT-Pairs, Wiki-Pairs, and PubMed-Pairs. As listed in Table 1, the details of the three datasets are as follows:

- NYT-Pairs are generated based on NYT corpus that is sampled from 2013 New York Times using Freebase (https://developers.google.com/freebase/) knowledge base and entity linker DBpedia Spotlight (Daiber et al., 2013).
- Wiki-Pairs are generated based on Wiki corpus that is sampled from Wikipedia using Freebase knowledge base and DBpedia Spotlight.
- PubMed-Pairs are generated based on PubMed corpus that is extracted from the paper abstracts of PubMed using UMLS (Bodenreider, 2004) knowledge base and entity linker PubTator (Wei et al., 2012).

Compared Approaches

The authors compare our approach with the following approaches:

- **Word2Vec-KNN:** A supervised approach, which uses the embeddings of Word2Vec (Mikolov et al., 2013) as input, and trains a nearest neighbor classifier for entity synonymous relation extraction.
- **Word2Vec-SVM:** A supervised approach, which uses the embeddings of Word2Vec as input, and trains an SVM classifier for entity synonymous relation extraction.
- **Word2Vec-BP:** A supervised approach, which uses the embeddings of Word2Vec as input, and trains a back propagation neural network classifier for entity synonymous relation extraction.
- **SynSetMine-IIP:** An instance-instance pair prediction approach (Shen et al., 2019), which uses the embeddings of Word2Vec as input, and trains a set-instance classifier for instance-instance synonymous relation pair extraction.
- **The Authors' Approach:** The authors' proposed approach, which uses the embeddings of Word2Vec as input, and trains a context-aware triplet network classifier for entity synonymous relation extraction.

Parameter Settings and Evaluation Metrics

For the sake of fair comparison, all the compared approaches use the Word2Vec embeddings released by Shen et al. (2019). For the embeddings of the entity relational contexts, the authors employ the skip-gram model (Mikolov et al., 2013) to train the embeddings on NYT, Wiki and PubMed, respectively. The authors use a five-fold validation method to tune the model parameters. The architecture of the

Table 1. Details of the datasets

Dataset	NYT-Pairs	Wiki-Pairs	PubMed-Pairs
#Documents	118,664	100,000	1,554,433
#Sentences	3,002,123	6,839,331	15,051,203
#Entities	1,670	4,046	3,229
#Positive pairs for training	736	1,517	2,926
#Negative pairs for training	1,472	3,034	5,852
#Positive pairs for testing	351	512	524
#Negative pairs for testing	702	1024	1,048

triplet network is five hidden layers fully connected neural network. The layer sizes of the triplet network are 300, 600, 1200, 600, and 300. The researchers use an Adam optimizer to optimize our proposed approach and set the initial learning rate to 0.001.

In order to evaluate the performance of each approach on entity synonymous relation identification, they report the macro average of precision (P), recall (R), and f1 score (F1), and present the precision-recall curves. In addition, the authors report the precision of top-N (P@N) entity synonymous relations produced by each approach.

Experimental Analysis

Overall Comparison

Table 2 shows the macro average of precision, recall, and f1 score for the compared approaches on three datasets. In general, it is possible to observe that the authors' approach with a context-aware triplet network classifier performs better than the other compared approaches in terms of precision, recall, and f1 score. The performance of Word2Vec-KNN on three datasets is much lower than the authors' approach. This means that a more refined classifier is needed to capture synonymous features from texts. Compared with Word2Vec-SVM, our approach has a significant improvement in recall and f1 score, but the improvement in precision is not significant. For example, the recall and f1 score of the authors' approach on WiKi-Pairs are 0.919 and 0.916, improved by 0.069 and 0.05 than Word2Vec-SVM, while the precision only improved by 0.02. For Word2Vec-BP and SynSetMine-IIP, the performance of SynSetMine-IIP is higher than Word2Vec-BP, but lower than the authors' approach. Although the set-instance classifier for SynSetMine-IIP is capable of capturing the permutation invariance information, it needs additional context information to improve the performance of entity synonymous relation identification. This demonstrates that the context-aware triplet network classifier in the authors' approach can capture more synonymous training signals for identifying entity synonymous relations.

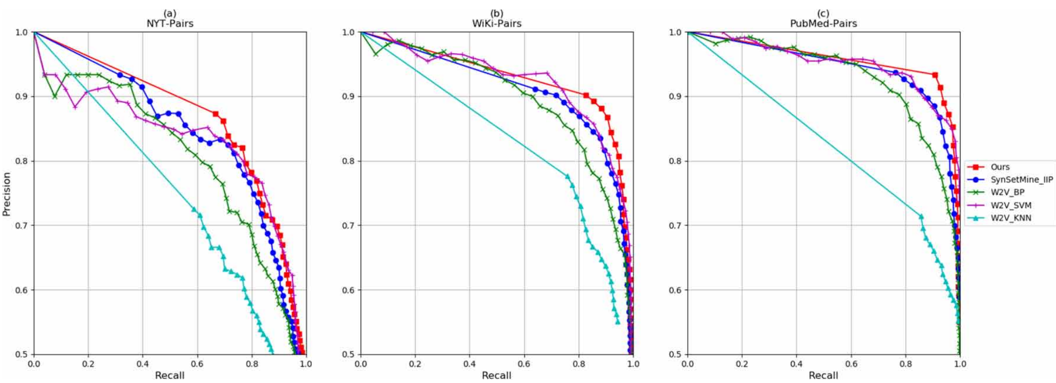
Figure 5 displays the precision-recall curves of the authors' approach and compared approaches. Compared with Word2Vec-KNN, Word2Vec-BP and SynSetMine-IIP, the precision of the authors' approach on three datasets performs better over the entire range of recall. Compared with Word2Vec-SVM, the authors' approach achieves higher precision on NYT-Pairs and PubMed-Pairs over the entire range of recall. An exception is that Word2Vec-SVM achieves slightly higher precision on WiKi-Pairs when the recall is range from 0.61 to 0.70. In general, it is possible to observe that the performances of the authors' approach outperform the compared approaches. It demonstrates that the context-aware triplet network classifier is beneficial for identifying entity synonymous relations.

Table 3 lists the precision values for the identified 100, 200, and 300 entity synonymous relations. These values highlight that the authors' approach performs better than the other compared approaches. Word2Vec-SVM and Word2Vec-BP achieve better precision than Word2Vec-KNN. This

Table 2. Precision, recall, and f1 score of the authors' approach and compared approaches

Competitors	NYT-Pairs			WiKi-Pairs			PubMed-Pairs		
	P	R	F1	P	R	F1	P	R	F1
Word2Vec-KNN	0.728	0.754	0.730	0.789	0.822	0.793	0.794	0.830	0.788
Word2Vec-SVM	0.837	0.775	0.794	0.894	0.850	0.866	0.917	0.919	0.918
Word2Vec-BP	0.791	0.804	0.797	0.868	0.854	0.860	0.888	0.877	0.883
SynSetMine-IIP	0.830	**0.838**	0.833	0.889	0.897	0.893	0.917	0.927	0.922
Ours	**0.852**	**0.838**	**0.844**	**0.914**	**0.919**	**0.916**	**0.936**	**0.945**	**0.940**

Figure 5. Precision-recall curves of the authors' approach and compared approaches



means that a more refined classifier can identify more valid entity synonymous relations. However, the performances of Word2Vec-SVM and Word2Vec-BP are lower than the authors' approach. For example, the P@200 of Word2Vec-SVM on WiKi-Pairs and the P@300 of Word2Vec-BP on PubMed-Pairs are 0.885 and 0.890, while those of the authors' approach are 0.935 and 0.947, improved by 0.050 and 0.057, respectively. Compared with SynSetMine-IIP, the authors' approach performs better on NYT-Pairs and WiKi-Pairs. The P@100 and P@300 of SynSetMine-IIP on PubMed-Pairs are equal to the authors' approach, but the P@200 is lower than the authors' approach. The above analysis illustrates the effectiveness of context-aware triplet network classifier on entity synonymous relation extraction task.

Ablation Study

In order to illustrate the effectiveness of entity relational contexts for entity synonymous relation identification, the authors implement a variant of their proposed approach named Ours-NoC, which uses the embeddings of Word2Vec as input, and trains a triplet network classifier without entity relational contexts. Tables 4 and 5 and Figure 6 show the experimental results of the authors' approach and ablation approach. They evidence that the performance of the authors' approach outperforms Ours-NoC. Especially, the f1 score and P@300 of the authors' approach on WiKi-Pairs are 0.916 and 0.920, improved by 0.023 and 0.04 than Ours-NoC, respectively. This again demonstrates that the entity relational contexts are beneficial for identifying entity synonymous relations.

Table 3. Precision values of the identified 100, 200, and 300 entity synonymous relations

Competitors	NYT-Pairs			WiKi-Pairs			PubMed-Pairs		
	P@100	P@200	P@300	P@100	P@200	P@300	P@100	P@200	P@300
Word2Vec-KNN	0.810	0.770	0.757	0.820	0.835	0.820	0.780	0.815	0.820
Word2Vec-SVM	0.880	0.865	0.860	0.880	0.885	0.860	0.910	0.920	0.920
Word2Vec-BP	0.860	0.855	0.843	0.870	0.880	0.857	0.880	0.900	0.890
SynSetMine-IIP	0.850	0.830	0.843	0.910	0.895	0.897	**0.930**	0.935	**0.947**
Ours	**0.900**	**0.875**	**0.873**	**0.950**	**0.935**	**0.920**	**0.930**	**0.940**	**0.947**

Table 4. Precision-recall curves of the authors' approach and ablation approach

Competitors	NYT-Pairs			WiKi-Pairs			PubMed-Pairs		
	P	R	F1	P	R	F1	P	R	F1
Ours-NoC	0.840	0.829	0.834	0.889	0.898	0.893	0.929	0.932	0.930
Ours	**0.852**	**0.838**	**0.844**	**0.914**	**0.919**	**0.916**	**0.936**	**0.945**	**0.940**

Analysis of Different Triplet Network Architectures

To further evaluate the effect of different triplet network architectures, the authors implement their approach with different hidden layer sizes. They denote the different hidden layer sizes as $\{x, 2{*}x, 4{*}x, 2{*}x, x\}$, where $x = \{100, 200, 300, 400, 500\}$. Figure 7 displays the macro average of precision, recall, and f1 score for the different triplet network architectures on three datasets. It is possible to observe that the performance of each architecture is growing as the x changes from 100 to 300. However, with the growth of hidden layer size, the performance of each architecture tends to be stable. Thus, the authors think that the size of the hidden layer in the range of 300 to 400 is enough to capture the synonymous signals for identifying entity synonymous relations.

Case Study

Finally, the authors present some extraction examples for the case study. As Table 6 illustrates, each dataset shows five examples, due to space constraints. The Table highlights that the authors' approach is capable of predicting most of the positive or negative entity synonymous relation pairs. For example, the prediction result of "planet earth" and "globe" is a positive pair, and "endocrine disease" and "quadriceps muscle" is a negative pair. However, there are still some wrong examples in the case study. For example, the ground truth of "spectacles" and "eyeglasses" is a positive pair, but the prediction result is a negative pair. This is because some words (e.g., "spectacles") rarely appear in the texts, so the authors' approach cannot capture enough synonymous signals to recognize entity synonymous relations.

CONCLUSION

In this paper, an entity synonymous relation extraction approach is proposed based on context-aware permutation invariance. The authors exploit a triplet network to learn the permutation invariance between the entities and integrate the entity relational contexts to discover entity synonymous relations. The main work is summarized as follows.

Figure 6. Precision-recall curves of the authors' approach and ablation approach

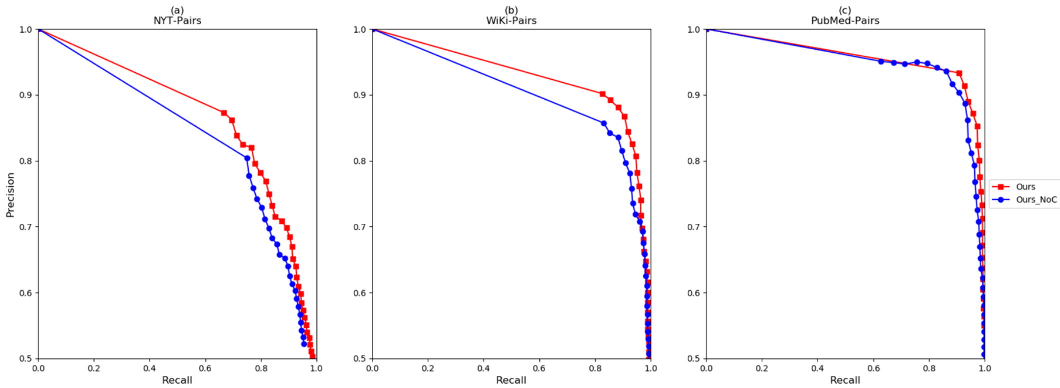
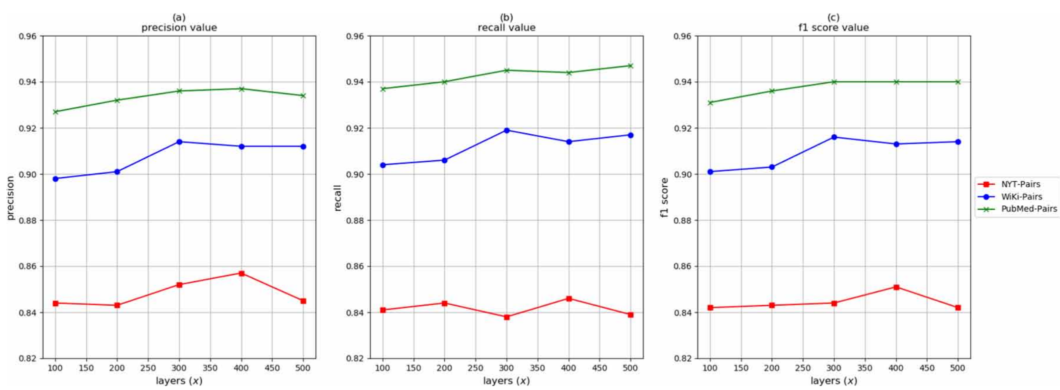


Table 5. P@N values of our approach and ablation approach

Competitors	NYT-Pairs			WiKi-Pairs			PubMed-Pairs		
	P@100	P@200	P@300	P@100	P@200	P@300	P@100	P@200	P@300
Ours-NoC	0.890	0.865	0.860	0.890	0.890	0.880	0.920	0.920	0.933
Ours	**0.900**	**0.875**	**0.873**	**0.950**	**0.935**	**0.920**	**0.930**	**0.940**	**0.947**

Figure 7. Analysis of different layer sizes for triplet network architecture



- An improved tripartite network framework is proposed to identify entity synonymous relations. The framework captures the permutation invariance between entities and determines whether two given entities possess synonymous relation.
- The relational context semantics and entity representations are integrated into the triplet network, which is capable of capturing more synonymous training signals to improve the performance of the triplet network framework in mining entity synonymous relations.
- The authors' approach is implemented on three real-world datasets. Experimental results illustrate that the authors' approach performs better than the other compared approaches on entity synonymous relation extraction task.

Table 6. Prediction examples of our approach on three datasets (√ denotes positive pair, and × denotes negative pair)

Datasets	Entity	Entity	Ground truth	Prediction
NYT-Pairs	Planet earth	Globe	√	√
	Phantom	Spectre	√	√
	Spectacles	Eyeglasses	√	×
	Laptop	Africans	×	×
	Baby	Newborn	√	√
WiKi-Pairs	United States federal government	U.S. government	√	√
	Motion picture	Movie	√	√
	Flower	Teen	×	×
	Heart disease	Disney	×	×
	Ladybird	Ladybugs	√	√
PubMed-Pairs	Booklets	Brochures	√	√
	Endocrine disease	Quadriceps muscle	×	×
	Tooth decay	Decayed teeth	√	√
	Antiemetic drugs	MALT lymphomas	×	×
	Urolith	Urinary calculi	√	√

In the future, the authors plan to apply their approach to Chinese real-world datasets, and would like to explore their approach in other research areas (e.g., hypernym-hyponym relation extraction).

ACKNOWLEDGMENT

The research reported in this paper was supported by the National Natural Science Foundation of China Youth Fund under Grant No. 61902001. This work was jointly supported by the Anhui Province University Outstanding Youth Talent Support Program under Grant No. gxyq2020031, and the Natural Science Project of Anhui Polytechnic University under Grant No. Xjky072019A01.

REFERENCES

Abu-Salih, B. (2021). Domain-specific knowledge graphs: A survey. *Journal of Network and Computer Applications*, *185*, 103076. doi:10.1016/j.jnca.2021.103076

Abu-Salih, B., Wongthongtham, P., & Chan, K. (2018). Twitter mining for ontology-based domain discovery incorporating machine learning. *Journal of Knowledge Management*, *22*(5), 949–981. doi:10.1108/JKM-11-2016-0489

Bodenreider, O. (2004). The unified medical language system (UMLS): Integrating biomedical terminology. *Nucleic Acids Research, 32*, 267-270.

Chakrabarti, K., Chaudhuri, S., Cheng, T., & Xin, D. (2012). A framework for robust discovery of entity synonyms. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1384-1392). ACM Press. doi:10.1145/2339530.2339743

Daiber, J., Jakob, M., Hokamp, C., & Mendes, P. N. (2013). Improving efficiency and accuracy in multilingual entity extraction. In *Proceedings of the 9th International Conference on Semantic Systems, ISEM* (pp. 121-124). ACM Press. doi:10.1145/2506182.2506198

Han, J., Pei, J., & Yin, Y. (2000). Mining frequent patterns without candidate generation. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data* (pp. 1-12). ACM Press.

Harris, Z. S. (1954). Distributional structure. *Word*, *10*(2-3), 146–162. doi:10.1080/00437956.1954.11659520

Huang, S., Luo, X., Huang, J., Guo, Y., & Gu, S. (2019). An unsupervised approach for learning a Chinese IS: A taxonomy from an unstructured corpus. *Knowledge-Based Systems*, *182*, 104861. doi:10.1016/j.knosys.2019.07.032

Huang, S., Luo, X., Huang, J., & Qin, W., & Gu, S. (2020). Neural entity synonym set generation using association information and entity constraint. In *Proceedings of the 2020 IEEE International Conference on Knowledge Graph (ICKG)* (pp. 321-328). IEEE Press. doi:10.1109/ICBK50248.2020.00053

Huang, S., Luo, X., Huang, J., Wang, H., Gu, S., & Guo, Y. (2020). Improving taxonomic relation learning via incorporating relation descriptions into word embeddings. *Concurrency and Computation*, *32*(14), e5696. doi:10.1002/cpe.5696

Jagannatha, A., Chen, J., & Yu, H. (2015). Mining and ranking biomedical synonym candidates from Wikipedia. In *Proceedings of the Sixth International Workshop on Health Text Mining and Information Analysis, Louhi@ EMNLP* (pp. 142-151). The Association for Computational Linguistics Press. doi:10.18653/v1/W15-2619

Kong, C., Gao, M., Xu, C., Fu, Y., Qian, W., & Zhou, A. (2019). Enali: Entity alignment across multiple heterogeneous data sources. *Frontiers of Computer Science*, *13*(1), 157–169. doi:10.1007/s11704-017-6561-3

Liu, W., Luo, X., Gong, Z., Xuan, J., Kou, N. M., & Xu, Z. (2016). Discovering the core semantics of event from social media. *Future Generation Computer Systems*, *64*, 175–185. doi:10.1016/j.future.2015.11.023

Lu, Y., & Hou, H. (2008). Research on automatic acquiring of Chinese synonyms from wiki repository. In *Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and International Conference on Intelligent Agent Technology-Workshops* (pp. 287-290). IEEE Computer Society Press.

Luo, X., Xu, Z., Yu, J., & Chen, X. (2011). Building association link network for semantic link on web resources. *IEEE Transactions on Automation Science and Engineering*, *8*(3), 482–494. doi:10.1109/TASE.2010.2094608

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *Proceedings of the 1st International Conference on Learning Representations, ICLR*.

Mintz, M., Bills, S., Snow, R., & Jurafsky, D. (2009). Distant supervision for relation extraction without labeled data. In *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the AFNLP* (pp. 1003-1011). The Association for Computer Linguistics Press. doi:10.3115/1690219.1690287

Nguyen, K. A., Walde, S. S., & Vu, N. T. (2017). Distinguishing antonyms and synonyms in a pattern-based neural network. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL* (pp. 76-85). The Association for Computer Linguistics Press. doi:10.18653/v1/E17-1008

Padmanabhan, D., Bhat, S., Shevade, S. K., & Narahari, Y. (2017). Multi-label classification from multiple noisy sources using topic models. *Information (Basel)*, *8*(2), 52. doi:10.3390/info8020052

Qu, M., Ren, X., & Han, J. (2017). Automatic synonym discovery with knowledge bases. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 997-1005). ACM Press. doi:10.1145/3097983.3098185

Roller, S., Kiela, D., & Nickel, M. (2018). Hearst patterns revisited: Automatic hypernym detection from large text corpora. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL* (pp. 358-363). The Association for Computer Linguistics Press. doi:10.18653/v1/P18-2057

Shen, J., Lyu, R., Ren, X., Vanni, M., Sadler, B. M., & Han, J. (2019). Mining entity synonyms with efficient neural set generation. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence* (pp. 249-256). AAAI Press. doi:10.1609/aaai.v33i01.3301249

Simanovsky, A., & Ulanov, A. (2011). Mining text patterns for synonyms extraction. In *Proceedings of the 2011 Database and Expert Systems Applications, DEXA, International Workshops* (pp. 473-477). IEEE Computer Society Press.

Sottovia, P., Paganelli, M., Guerra, F., & Velegrakis, Y. (2019). Finding synonymous attributes in evolving Wikipedia infoboxes. In *Proceedings of the Advances in Databases and Information Systems - 23rd European Conference, ADBIS* (vol. 11695, pp. 169-185). Springer Press. doi:10.1007/978-3-030-28730-6_11

Turney, P. D. (2001). Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In *Proceedings of the Machine Learning: EMCL 2001, 12th European Conference on Machine Learning* (*vol. 2167*, pp. 491-502). Springer Press.

Wang, C., Fan, Y., He, X., & Zhou, A. (2019). Predicting hypernym-hyponym relations for Chinese taxonomy learning. *Knowledge and Information Systems*, *58*(3), 585–610. doi:10.1007/s10115-018-1166-1

Wang, H., Ren, H., & Leskovec, J. (2020). *Entity context and relational paths for knowledge graph completion.* CoRR, abs/2002.06757.

Wang, W., Thomas, C., Sheth, A. P., & Chan, V. (2010). Pattern based synonym and antonym extraction. In *Proceedings of the 48th Annual Southeast Regional Conference* (pp. 64). ACM Press.

Wei, C., Harris, B. R., Li, D., Berardini, T. Z., Huala, E., Kao, H., & Lu, Z. (2012). Accelerating literature curation with text-mining tools: A case study of using Pubtator to curate genes in Pubmed abstracts. *Database the Journal of Biological Databases & Curation, 2012*, bas041.

Wongthongtham, P., & Abu-Salih, B. (2018). Ontology-based approach for identifying the credibility domain in social Big Data. *Journal of Organizational Computing and Electronic Commerce*, *28*(4), 354–377. doi:10.1080/10919392.2018.1517481

Xie, P., Yang, D., & Xing, E. P. (2015). Incorporating word correlation knowledge into topic modeling. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT* (pp. 725-734). The Association for Computational Linguistics Press. doi:10.3115/v1/N15-1074

Yin, D., Hu, Y., Tang, J., Zhou, M., Ouyang, H., Kang, C., Deng, H., Nobata, C., Langlois, J., & Chang, Y. (2016). Ranking relevance in Yahoo search. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 323-332). ACM Press.

Nan Yan is an associate professor in the School of Computer and Information, Anhui Polytechnic University, Wuhu, China. He received his master's degree in University of Electronic Science And Technology Of China, 2012. His research interests include software defined network, reinforcement learning, and data mining.

Subin Huang is a lecturer in the School of Computer and Information, Anhui Polytechnic University, Wuhu, China. He received the master's degree in School of Computer and Information, Anhui Polytechnic University, 2012, China, and received the Ph.D. degrees in the School of Computer Engineering and Science, Shanghai University, China. His main research interests include information retrieval, data mining, and knowledge graph. Chao Kong received his Ph.D. degree in software engineering from the Institute for Data Science and Engineering, East China Normal University, Shanghai, in 2017. He is a lecture at the School of Computer and Information at Anhui Polytechnic University (AHPU), Wuhu. His research interests include web data management, streaming data processing, social network analysis, and data mining.