Cross-Lingual Entity Matching for Heterogeneous Online Wikis

Weiming Lu(✉), Peng Wang, Huan Wang, Jiahui Liu, Hao Dai, and Baogang Wei

College of Computer Science and Technology,
Zhejiang University, Zhejiang, China
luwm@zju.edu.cn

Abstract. Knowledge bases play an increasing important role in many applications. However, many knowledge bases mainly focus on English knowledge, and have only a few knowledge for low-resource languages (LLs). If we can map the entities in LLs to these in high-resource languages (HLs), many knowledge such as relation between entities can be transferred from HLs to LLs.

In this paper, we propose an efficient and effective Cross-Lingual Entity Matching approach (CL-EM) to enrich the existing cross-lingual links by learning to rank framework with the learned language-independent features, including cross-lingual topic features and document embedding features. In the experiments, we verified our approach on the existing cross-lingual links between Chinese Wikipedia and English Wikipedia by comparing it with other state-of-art approaches. In addition, we also discovered 141,754 new cross-lingual links between Baidu Baike and English Wikipedia, which almost doubles the number of the existing cross-lingual links.

1 Introduction

Knowledge bases play an increasingly important role in many applications such as information retrieval, machine translation, and question answering. However, many knowledge bases mainly focus on English knowledge, and have only a few knowledge for low-resource languages (LLs). For example, DBPedia [8] contains 4.68 million entities in English, but only contains 0.78 millions entities in Chinese[1]. On the other hand, there are two large-scale Chinese encyclopedias named Baidu Baike[2] and Hudong Baike[3], which both have more than 12 millions articles. But these is no mature Chinese knowledge base. Therefore, if we can map the entities in LLs to the entities in high-resource languages (HLs), many knowledge (i.e. relations between entities) can be transferred from HLs to LLs.

[1] http://wiki.dbpedia.org/dbpedia-2016-04-statistics.
[2] http://baike.baidu.com.
[3] http://www.baike.com.

X. Huang et al. (Eds.): NLPCC 2017, LNAI 10619, pp. 887–899, 2018.
https://doi.org/10.1007/978-3-319-73618-1_78

In this paper, we try to address the entity matching problem in the cross-lingual environment, especially for the Chinese and English entities, but the task is not trivial. The main challenges are as follows: (1) Different languages are used to describe cross-lingual entities, so the similarity between them can not be calculated directly since they are in different word space. Can we find some language-independent features for cross-lingual entity matching? (2) Millions of articles exist in both Wikipedia and Baidu Baike. How to develop an efficient and effective approach to deal with such large-scale data sets?

In order to solve the above challenges, we take full advantage of the limited but useful English-Chinese cross-lingual links within Wikipedia. We first use them to generate candidates for reducing the computation, and then train a cross-lingual topic model and a cross-lingual document representation model to extract the language-independent features.

Our contributions are as follows: (1) We propose an efficient and effective Cross-Lingual Entity Matching approach (CL-EM) to enrich the existing cross-lingual links by learning to rank with some language-independent features. (2) We evaluate our approach on the existing cross-lingual links in Wikipedia. In practice, we can find the corresponding cross-lingual entity in Wikipedia or NIL for entities in Baidu Baike.

2 Related Work

2.1 Entity Matching

Currently, most works focus on monolingual entity matching tasks such as SIGMa [6], LINDA [2] and PARIS [14] by utilizing the structural information of RDF triples (subject, predicate, object) in knowledge bases. For example, SIGMa [6] presented a Simple Greedy Matching algorithm for aligning knowledge bases with an iterative propagation procedure. LINDA [2] is also an iterative greedy algorithm for entity matching by using prior similarity and contextual similarity from its neighboring entities. PARIS [14] computed alignments not only for entities, but also for classes and relations based on a probabilistic framework.

However, Chinese encyclopedias such as Baidu Baike only contain raw articles as in Wikipedia. Therefore, the traditional approaches mentioned above are not feasible for the cross-lingual entity matching between Baidu Baike and English Wikipedia, since they use RDF triples or ontologies in matching. While our approach does not rely on RDF triples, and only uses language independent features of articles.

Crowdsourcing also has attracted significant attention in entity resolution (e.g., [15–17]). However, crowdsourcing is not the focus of our paper.

2.2 Cross-Lingual Links Discovery

The most related work is to discover cross-lingual links within Wikipedia. BabelNet [11] and YAGO3 [10] both aim to build a multilingual knowledge base,

but they mainly relied on machine translation to discover cross-lingual links. In addition, [9] also used machine translation to interlink documents described in English and Chinese languages.

Besides machine translation based approaches, [13] tackled the cross-lingual links discovery between German and English Wikipedia using a classification-based approach. They designed several features include chain link count feature, text features and graph features. However, the text features based on text overlap and similarity are strong features with good classification results according to their experiments, so the approach may not be suitable for other language pairs, such as English and Chinese.

The existing cross-lingual links of Wikipedia are also widely used. For example, [18] proposed a factor graph based approach with link-based features to predict new cross-lingual links in Wikipedia. [12] proposed Cross-Language Explicit Semantic Analysis (CL-ESA) by using the existing cross-language links to represent documents in different languages as vectors for cross-lingual information retrieval.

3 Problem Formulation

In this section, we formally define the cross-lingual entity matching problem.

In encyclopedias such as Wikipedia and Baidu Baike, each article can be represented as a seven-tuple $x = \{tl, abs, txt, clg, tags, ilnk, olnk\}$, where tl, abs, clg and txt are title, abstract, catalog and content of the article x, $tags$, $ilnk$ and $olnk$ are the sets of category tags, inlinks and outlinks of x. Therefore, the cross-lingual entity matching problem can be defined as follows:

Problem (Cross-lingual Entity Matching). *Given two encyclopedias* $\mathcal{X}_1 = \{x_i^1 \,|\, i = 1, 2, \ldots, M\}$ *and* $\mathcal{X}_2 = \{x_i^2 \,|\, i = 1, 2, \ldots, N\}$ *in different languages (e.g.* $\mathcal{X}_c = \{x_i^c\}$ *and* $\mathcal{X}_e = \{x_i^e\}$ *are encyclopedias in Chinese and English respectively), the goal of the cross-lingual entity matching is to find, for each article* $x_i^1 \in \mathcal{X}_1$*, an equivalent article* $x_i^2 \in \mathcal{X}_2$ *or NIL if there is no equivalent entity in* $\mathcal{X}_2$*.*

4 Cross-lingual Entity Matching

In this section, we will describe the cross-lingual entity matching approach in detail. We first give an overview of the approach, and then elaborate the candidate selection, feature extraction and candidate ranking respectively.

Figure 1 shows the overview of the approach. When given an query entity $x_i^1 \in \mathcal{X}_1$, we first generate a set of candidate entities $\mathcal{C}(x_i^1) = \{x_j^2 \in \mathcal{X}_2\}_{j=1}^{|C|}$ from $\mathcal{X}_2$ to reduce the complexity of entity matching, and then extract features such as handcrafted feature, topic feature and document embedding for each entity pair $(x_i^1, x_j^2 \in \mathcal{C}(x_i^1))$. Finally, the equivalent entity from $\mathcal{C}$ (or NIL) is selected within the ranking layer.

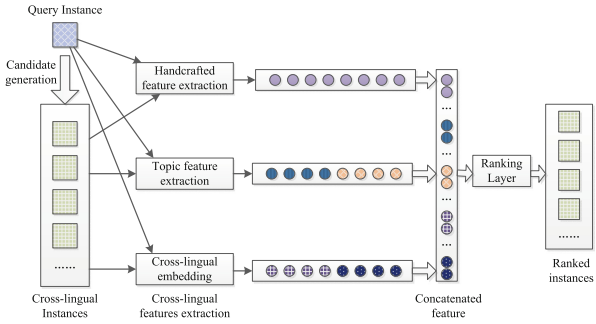

Fig. 1. The overview of the cross-lingual entity matching

4.1 Candidate Selection

It is time consuming to select the equivalent entity from millions of entities when given a query entity, so we use a candidate selection strategy to reduce the complexity.

According to the *chain link hypothesis* [13], given an article $x_i^1 \in \mathcal{X}_1$, if there is a *chain link* between x_i^1 and $x_j^2 \in \mathcal{X}_2$, then x_j^2 could be one of the equivalent entity candidates. Formally, the *chain link* between x_i^1 and x_j^2 is defined as: $x_i^1 \rightarrow x_p^1 \equiv x_q^2 \leftarrow x_j^2$, where x_i^1 and x_q^1 are articles in $\mathcal{X}_1$, x_j^2 and x_q^2 are articles in $\mathcal{X}_2$. $x_p^1 \equiv x_q^2$ means there is a cross-lingual link between x_p^1 and x_q^2, and $x_i^1 \rightarrow x_p^1$ means there is a inner-link in x_i^1 pointing to x_p^1. We denote the equivalent entity candidate of x_i^1 as $\mathcal{C}_{all}(x_i^1) = \{x_j^2 \,|\, \exists x_p^1 \in \mathcal{X}_1, \exists x_q^2 \in \mathcal{X}_2, x_i^1 \rightarrow x_p^1 \equiv x_q^2 \leftarrow x_j^2\}$.

However, there are still many candidate entities in $\mathcal{C}_{all}(x_i^1)$, so we should reduce the $\mathcal{C}_{all}(x_i^1)$ further. Obviously, if there are more chain links between two entities, these two entities are more likely to be equivalent. In order to verify this assumption, we randomly selected 3000 Chinese articles with the existing cross-lingual links to the English articles from Wikipedia, denoted as $\{(x_i^c, x_i^e) \,|\, x_i^c \in \mathcal{X}_c, x_i^e \in \mathcal{X}_e, x_i^c \equiv x_i^e, i = 1, 2, \ldots, 3000\}$, and generated the candidate entity set $\mathcal{C}_n(x_i^c)$ for each article x_i^c, which has top n candidate entities ranked by the number of *chain links*. Then, we checked whether x_i^e is in the $\mathcal{C}_n(x_i^c)$, and got $Pr(x_i^e \in \mathcal{C}_{1000}(x_i^c)) = 79.67\%$, $Pr(x_i^e \in \mathcal{C}_{5000}(x_i^c)) = 86.03\%$, and $Pr(x_i^e \in \mathcal{C}_{all(57447)}(x_i^c)) = 94.17\%$. We find that most of the equivalent entities are in the candidate set $\mathcal{C}_{all}$, and $n = 1000$ is a good trade-off between the precision and the complexity. Therefore, we selected the equivalent entity for x_i^1 from the candidate set $\mathcal{C}_{1000}(x_i^1)$ in the following sections.

4.2 Feature Extraction

In order to select the equivalent entity for x_i^1 from the candidate set $\mathcal{C}_{1000}(x_i^1)$, we should calculate the equivalence score for each pair $(x_i^1, x_j^2 \in \mathcal{C}_{1000}(x_i^1)$, and then rank them according to the scores. The features used in the ranking procedure include three types: handcraft features, cross-lingual topic feature and document embedding feature.

Handcraft Features. Intuitively, if two articles x_i^1 and x_j^2 have similar titles through translation, and they link to several equivalent entities and categories, then they would likely to be equivalent. Thus, we define eight handcraft features as follows.

Feature **1** (Title Similarity)
We translate the title of article x_i^c from Chinese to English, and then calculate the edit distance between two titles as the feature: $f_1 = edit\_distance(TransC2E(x_i^c.tl), x_j^e.tl)$.

If two articles have equivalent inlinks, outlinks and categories, they tend to be equivalent, which has been adequately proven in [18], so we design the 2^{nd} to 8^{th} features based on the existing cross-lingual links as follows.
Feature **2** (Outlink Overlap): $f_2 = |\{(a,b) \,|\, a \equiv b, a \in x_i^c.olnk, b \in x_j^e.olnk\}|$
Feature **3** (Jaccard Coefficient Of Outlink): $f_3 = f_2/(|x_i^c.olnk| + |x_j^e.olnk| - f_2)$
Feature **4** (Normalized Outlink Overlay): $f_4 = f_2/(\max(|\{(a,b) \,|\, a \equiv b, a \in x_i^c.olnk, b \in x_j^e.olnk, x_j^e \in \mathcal{C}(x_i^c)\}|))$
Feature **5** (Inlink Overlap): $f_5 = |\{(a,b) \,|\, a \equiv b, a \in x_i^c.ilnk, b \in x_j^e.ilnk\}|$
Feature **6** (Jaccard Coefficient Of Inlink): $f_6 = f_3/(|x_i^c.ilnk| + |x_j^e.ilnk| - f_5)$
Feature **7** (Tags Overlap): $f_7 = |\{(a,b) \,|\, a \equiv b, a \in x_i^c.tags, b \in x_j^e.tags\}|$
Feature **8** (Jaccard Coefficient Of Tags): $f_8 = f_3/(|x_i^c.tags| + |x_j^e.tags| - f_7)$

Finally, the handcraft features of two articles x_i^1 and x_j^2 can be represented as $v_h(x_i^1, x_j^2) = [f_1, f_2, \ldots, f_8]$.

Cross-lingual Topic Model. If two articles x_i^1 and x_j^2 are equivalent even in different languages, they must have similar topic distribution.

In order to represent the articles in both $\mathcal{X}_1$ and $\mathcal{X}_2$ using the same topic set, we learn a topic model for $\mathcal{X}_1$ and $\mathcal{X}_2$ simultaneously. We first construct the *pseudo document* by concatenating the abstracts and catalogs of two equivalent articles as $d_i = \{x_i^1.abs \cup x_i^1.clg \cup x_i^2.abs \cup x_i^2.clg \,|\, x_i^1 \equiv x_i^2, x_i^1 \in \mathcal{X}_1, x_i^2 \in \mathcal{X}_2\}$. Then, we apply LDA (Latent Dirichlet Allocation) [1] on the *pseudo document set* $D = \{d_i\}$ to learn the cross-lingual topic model. With this model, we can map article x_i in both $\mathcal{X}_1$ and $\mathcal{X}_2$ into a topic distribution vector $v_t(x_i)$ with the same topic set.

In our experiment, we generated the *pseudo document set* from 100,000 article pairs in Wikipedia, and some examples of topics generated by our approach are shown in Table 1, where top 5 terms in two languages ranked by the probability in three topics are listed.

From the table, we find that LDA can conceptually cluster highly similar terms into the same topics, even they are in different languages. Based on the cross-lingual topic model, articles in both $\mathcal{X}_1$ and $\mathcal{X}_2$ can be represented with the same topic sets.

Table 1. Examples of topics generated by cross-lingual topic model

Topic 1		Topic 2		Topic 3	
Chinese	English	Chinese	English	Chinese	English
电影(film)	film	位于(located at)	city	运动员(athlete)	team
美国(American)	series	平方公里(square km)	area	效力(play for)	season
作品(production)	music	人口(population)	river	冠军(champion)	league
日本(Japanese)	new	面积(area)	town	球队(team)	club
导演(director)	release	城市(city)	population	比赛(race)	cup

Cross-lingual Document Embedding. Recently, representation learning methods such as Paragraph Vectors [7] have been proposed to learn continuous distributed vector representations for pieces of texts, and outperform other document modeling algorithms like LDA [3]. However, articles in $\mathcal{X}_1$ and $\mathcal{X}_2$ represented by Paragraph Vectors are in different language spaces, they can not be compared with each other directly. Therefore, we learn the cross-lingual document embedding vector for every article in both $\mathcal{X}_1$ and $\mathcal{X}_2$ with a deep rank model based on the Paragraph Vectors.

Suppose article x_i is represented as $f(x_i)$ in the embedding space, then the similarity between two articles x_i and x_j is measured by: $D(f(x_i), f(x_j)) = ||f(x_i) - f(x_j)||_2^2$. The smaller the distance $D(x_i, x_j)$ is, the more similar between the article x_i and x_j are. For a triplet $t_q = (x_q^1, x_p^2, x_n^2)$, where $x_q^1 \equiv x_p^2$ and $x_n^2 \in \mathcal{C}_{1000}(x_q^1)/x_p^2$, we can define the loss as: $l(x_q^1, x_p^2, x_n^2) = max\{0, g + D(f(x_q^1), f(x_p^2)) - D(f(x_q^1), f(x_n^2))\}$, where g is a margin parameter. Finally, our objective function is:

$$\min \sum_{q \in Q} \xi_q + \lambda ||W||_2^2$$
$$\text{s.t. } max\{0, g + D(f(x_q^1), f(x_p^2)) - D(f(x_q^1), f(x_n^2))\} < \xi_q$$
$$\forall x_q^1, x_p^2, x_n^2 \text{ such that } x_q^1 \equiv x_p^2, x_n^2 \in \mathcal{C}_{1000}(x_q^1)/x_p^2$$

where W is the parameters of $f(\cdot)$, λ is the parameter to improve the generalization, and Q is the training data. The neural network of the deep ranking model is shown in the Fig. 2, which includes three full-connected layers and a local normalized layer, and then a ranking layer on the top evaluates the hinge loss of a triplet.

Training a deep neural network usually needs a large amount of training data, thus we randomly select 100,000 articles $Q = \{x_q^1 | \exists x_p^2 \equiv x_q^1, x_q^1 \in \mathcal{X}_1, x_p^2 \in \mathcal{X}_2\}$, and then form ten triplets $\{x_q^1, x_p^2, x_n^2\}$ for each $x_q^1 \in Q$ as following: At first, we select the equivalent article $x_p^2 \in \mathcal{X}_2$ for x_q^1, and then select five articles from $\mathcal{C}_{1000}(x_q^1)/x_p^2$ and five articles from $\mathcal{X}_2/\mathcal{C}_{1000}(x_q^1)$ respectively as the x_n^2. In our experiment, we used TensorFlow[4] to train our model, and parameters were set with $\lambda = 0.1$, $g = 0.8$, batch size $= 400$ and learning rate $= 0.001$.

[4] https://www.tensorflow.org/.

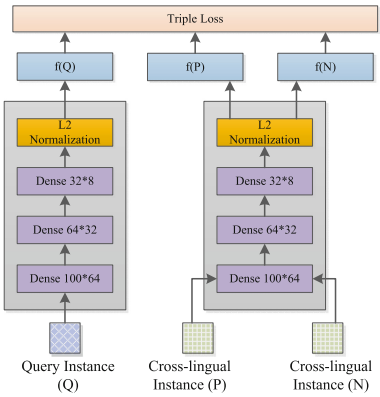

Fig. 2. Cross-lingual documents embedding

4.3 Candidate Ranking

Given an article $x_i^1 \in \mathcal{X}_1$, we want to select the equivalent article (or NIL) from $\mathcal{C}_{1000}(x_i^1)$. In this section, we model the problem as a *learning to rank* problem.

We apply RankSVM [5] as the ranking model, and take the features extracted in the previous sections as the input. Formally, the training data set is denoted as $S = \{(x_i^1, \mathcal{C}_{1000}(x_i^1)), \mathbf{y}_i\}_{i=1}^m$, where a feature vector for an article pair $(x_i^1, x_j^2 \in \mathcal{C}_{1000}(x_i^1))$ is created by concatenating the extracted features $v_{ij} = v_a(x_i^1, x_j^2) = [v_h(x_i^1, x_j^2), v_t(x_i^1), v_t(x_j^2), f(x_i^1), f(x_j^2)]$, $\mathbf{y}_i = [y_{i,1}, y_{i,2}, \ldots, y_{i,|\mathcal{C}_{1000}(x_i^1)|}]$, and $y_{i,j} = 1$ when $x_i^1 \equiv x_j^2$, otherwise $y_{i,j} = 0$.

With the ranking model, we can rank the articles in $\mathcal{C}_{1000}(x_i^1)$ for x_i^1 according to their relevance score $y_{i,j}$. However, there may be no equivalent article in $\mathcal{C}_{1000}(x_i^1)$ to x_i^1 in reality. That is, the article with the highest relevance score may not be the equivalent article to x_i^1, or these is even no equivalent article in the ranking candidates. Therefore, we define two thresholds to disambiguate the *NIL* and the equivalent article, inspired by [19]: $t_1 = h - margin_1 \cdot (h - avg)$ and $t_2 = margin_2 \cdot (h - avg)$, where h, s and avg are the highest, second-highest and average scores in $\mathbf{y}_i$, $margin_1$ and $margin_2$ are two margin parameters, which are determined in the experiments. If $s < t_1$, the article with the highest score can be considered as the equivalent article, and if $s > t_2$, there would be no equivalent article.

5 Experiments

5.1 Datasets

We constructed two datasets from Wikipedia:

Dataset **1**: As in [18], we randomly selected 2000 Chinese Wikipedia articles with existing cross-lingual links to English articles, denoted as D, and then picked out the corresponding 2000 English articles to form 2000 cross-lingual

article pairs. Here, 2000 English articles are considered as the candidate articles $\mathcal{C}_{2000}(x_i^c)$ for each Chinese article $x_i^c \in \mathcal{X}_c$.

Dataset **2**: Similar to the Dataset **1**, we randomly selected 3000 Chinese Wikipedia articles with existing cross-lingual links to English articles, denoted as D, but we used the proposed candidate selection method to generate $\mathcal{C}_{1000}(x_i^c)$ for each Chinese article $x_i^c \in \mathcal{X}_c$ from all English Wikipedia articles. Then, we checked whether the equivalent article $x_i^e \in \mathcal{X}_e$ for x_i^c exists in $\mathcal{C}_{1000}(x_i^c)$. Only 2390 Chinese articles have its equivalent articles in the candidate set, so we used this 2390 Chinese articles with its equivalent articles as the Dataset **2**.

Obviously, it is more challenging when evaluating the approaches with the Dataset **2**, since the articles in the candidate set are very similar.

For each dataset, we used 75% of the data as the training data, and the remaining data as the testing data.

5.2 Comparison Methods

We compared our approach with the following methods:

- **Title Match (TM).** We translate the title of Chinese article into English through Baidu Translate API[5], and then match the title with English articles in the candidate set to check whether they are exactly same.
- **Title Similarity (TS).** Similar to TM, TS considers the article with the minimal edit distance between the translated title and the title of each English article in the candidate set as the equivalent article.
- **Support Vector Machine (SVM).** We used handcrafted feature v_h as the input, and trained SVM classifiers on the Dataset **1** and Dataset **2** respectively.
- **Similarity Aggregation (SA).** Here, we considered the average similarity of some handcrafted features as the article similarity. Thus, for each Chinese article, we select the most similar English article as the equivalent article. In order to evaluate the influence of the *Title Similarity*, we calculated the article similarity in two ways: $SA_1(x_i^c, x_j^e) = (f_1 + f_3 + f_6 + f_8)/4$ and $SA_2(x_i^c, x_j^e) = (f_3 + f_6 + f_8)/3$.
- **Cross-Language Explicit Semantic Analysis (CL-ESA)** [12]**.** CL-ESA is the cross-lingual extension to the Explicit Semantic Analysis (ESA) approach [4]. Here, we used the terms having existing cross-lingual links to represent articles in both $\mathcal{X}_c$ and $\mathcal{X}_e$.

In the experiments, we used precision, recall and F-score as the evaluation metrics.

5.3 Results

In this section, we only evaluate the performance of the candidate ranking, and don't predict the exactly equivalent article and NIL by comparing the ranking scores to the thresholds t_1 and t_2, which will be evaluated in the next section.

[5] http://api.fanyi.baidu.com.

Table 2. Results on DATASET **1**

Methods	Prec.	Rec.	F_1
TM	100.00%	24.55%	39.42%
TS	59.30%	59.30%	59.30%
SA_1	85.35%	85.35%	85.35%
SA_2	82.00%	82.00%	82.00%
SVM	75.20%	92.50%	82.80%
CL-EM (v_h)	92.30%	92.30%	92.30%
CL-EM (v_a)	92.50%	92.50%	**92.50%**

Table 3. Results on DATASET **2**

Methods	Prec.	Rec.	F_1
TM	97.65%	26.03%	41.10%
TS	56.03%	56.03%	56.03%
SA_1	65.60%	65.60%	65.60%
SA_2	34.73%	34.73%	34.73%
CL-ESA	7.3%	7.3%	7.3%
CL-EM (v_h)	68.75%	68.75%	68.75%
CL-EM ($v_h + v_t$)	69.44%	69.44%	69.44%
CL-EM (v_a)	70.28%	70.28%	**70.28%**

Since TM and SVM try to predict whether x_i^c and $x_j^e \in \mathcal{C}(x_i^c)$ is equivalent, so it is possible that none of the article in $\mathcal{C}(x_i^c)$ is the equivalent article. While for TS, SA, CL-ESA and our approach, they rank the articles in the candidate set, and consider the Top 1 as the equivalent article, so they have the same recall and precision. The comparison results of DATASET **1** are shown in the Table 2.

From the table, we can see that (1) Since TM is based on the exact title matching, so the precision reaches to 100%, but it has a very low recall because of improper translation. (2) TS increases the recall by ranking the candidate articles according to the title similarity, but it decreases the precision. (3) SVM and SA both considered all the handcraft features, so their F_1 scores are larger than 80%. Especially, the F_1 scores of SA and SVM are larger than that of TS, which indicates that the in-links, out-links and tags are very useful in cross-lingual entity matching. In addition, the results between SA_1 and SA_2 indicates the usefulness of title similarity in cross-lingual entity matching. (4) Our approach CL-EM outperforms other methods significantly. When only using the handcraft features, CL-EM can reach the F_1 score 92.3% straightforwardly, but only 0.2% can be improved when adding cross-lingual topic features and document embedding features. This may be because the articles in DATASET **1** are quit different, the handcraft features can be adequate to distinguish them well.

For the more challenging dataset DATASET **2**, we obtained a worse performance than that in the DATASET **1**. The details are shown in the Table 3.

From the table, we can see that CL-EM still outperforms all other methods significantly. Since many articles have the same title, but refer to different entities in the real world, so the precision of TM doesn't reach to 100%. In addition, cross-lingual topic feature and article embedding feature can indeed improve the cross-lingual entity matching by comparing CL-EM(v_h), CL-EM($v_h + v_t$) and CL-EM(v_a). Surprisedly, we only obtain 7.3% for CL-ESA, because the articles in the candidate set are very similar.

In addition, we also evaluate the performance of CL-EM according to the Top-K precision by: $prec_k = \frac{\sum_{x_i^c \in D} |\delta(x_i^e \in TopK(\mathcal{C}_{1000}(x_i^c)) \mid x_i^e \equiv x_i^c)|}{|D|}$, where $TopK(\mathcal{C}_{1000}(x_i^c))$ is the top k articles in the candidate set $\mathcal{C}_{1000}(x_i^c)$ for article

x_i^c according to the ranking score. $\delta(true) = 1$ and $\delta(false) = 0$. Table 4 shows the results. Obviously, the precision increases along with the larger k. Indeed, most of the equivalent articles are ranked in the Top k list. Thus, in our practical system, we show the Top k list to users, and users can select and click the equivalent cross-lingual article. With this user crowdsourcing activities, we can improve the quality of cross-lingual entity matching.

Table 4. Evaluation for the Top K articles in the candidate set

k	1	2	5	10
$prec_k$	70.28%	75.28%	81.81%	86.25%

When training the cross-lingual document embedding model, different margin parameter g would influence the model. Thus, we evaluate the model with different g and the different ways of concatenating two document embedding vectors. The results are shown in Table 5, where v_1v_2, $|v_1 - v_2|$ and $||v_1 - v_2||_2$ are three ways to combine two vector $v_1 = f(x_i^1)$ and $v_2 = f(x_j^2)$. v_1v_2 is to concatenate two vectors, $|v_1 - v_2|$ is a N-dimensional vector $< |v_{1_1} - v_{2_1}|, |v_{1_2} - v_{2_2}|, \ldots, |v_{1_N} - v_{2_N}| >$ and $||v_1 - v_2||_2$ is the Euclidean distance of two vectors. According to the result, we chose $g = 0.8$ and $||v_1 - v_2||_2$ in the experiments since they reach the best performance.

Table 5. The F_1 of CL-EM with different settings in cross-lingual document embedding

g	v_1v_2	$\|v_1 - v_2\|$	$\|\|v_1 - v_2\|\|_2$
0.1	68.75%	68.47%	68.47%
0.5	68.61%	68.61%	69.86%
0.8	68.33%	69.03%	**70.28%**
1.0	68.61%	69.58%	68.19%

5.4 Equivalence Judgement Evaluation

In this section, we evaluate the performance of equivalence judgement by turning the parameters $margin_1$ and $margin_2$. In the equivalence judgement, we not only need to judge exactly whether the article with the highest relevance score in $\mathcal{C}(x_i^1)$ is equivalent to the article x_i^1 (Task 1), but also need to judge whether there is no equivalent article in $\mathcal{C}(x_i^1)$ for the article x_i^1 (Task 2). Therefore, we constructed two datasets Dataset_Top1 and Dataset_NIL from Baidu Baike and English Wikipedia.

Dataset_Top1: This dataset is used for Task 1. We randomly selected 300 positive samples $P = \{x_i^c \,|\, Top(\mathcal{C}(x_i^c)) = x_i^e \&\& x_i^c \equiv x_i^e, i = 1, 2, \ldots, 300\}$, and then selected 300 negative samples $N = \{x_i^c \,|\, Top(\mathcal{C}(x_i^c)) \neq x_i^e \&\& x_i^c \equiv x_i^e, i = 1, 2, \ldots, 300\}$. Here, $Top(\mathcal{C}(x_i^c))$ is the article with the highest relevance score in $\mathcal{C}(x_i^c)$ for x_i^c.

DATASET_NIL: This dataset is used for Task 2. We randomly selected 600 positive samples $P = \{x_i^c \,|\, x_i^e \notin \mathcal{C}(x_i^c) \&\& x_i^c \equiv x_i^e, i = 1, 2, \ldots, 600\}$, and then selected 600 negative samples $N = \{x_i^c \,|\, Top(\mathcal{C}(x_i^c)) = x_i^e \&\& x_i^c \equiv x_i^e, i = 1, 2, \ldots, 300\} \cup \{x_i^c \,|\, x_i^e \in \mathcal{C}(x_i^c) \&\& x_i^c \equiv x_i^e, i = 1, 2, \ldots, 300\}$.

We assume Q is the equivalence judgement results. For task 1, $Q = \{x_i^c \,|\, s < t_1\}$ and $Q = \{x_i^c \,|\, s > t_2\}$ for Task 2, where s, t_1 and t_2 are defined in Sect. 4.3. Then, the precision and recall are calculated as: $p = |Q \cap P|/|Q|$ and $r = |Q \cap P|/|P|$. Since precision is more important than recall, so we also calculated $F_{0.5}$ as $\frac{1.25 \cdot (p+r)}{0.25 \cdot p + r}$. The results are shown in Figs. 3 and 4.

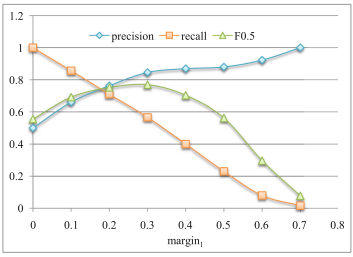

Fig. 3. performance for Top 1 equivalence judgement varies with parameter $margin_1$ in DATASET_TOP1

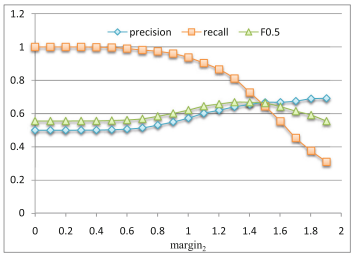

Fig. 4. performance for NIL detection varies with parameter $margin_2$ in DATASET_NIL

From the figures, we can see that precision increases when $margin_1$ and $margin_2$ get larger, but recall decreases. Therefore, we selected $margin_1 = 0.3$ and $margin_2 = 1.3$ to discover cross-lingual links between Baidu Baike and English Wikipedia.

Finally, we used our approach to discover new cross-lingual links between Baidu Baike and English Wikipedia. We crawled 10,143,321 articles from Baidu Baike, and then extracted 407,092 cross-lingual links of articles and 82,452 cross-lingual links of categories which already exist between Chinese Wikipedia and English Wikipedia. Then we obtained 173,259 equivalent articles in Baidu Baike among these existing cross-lingual links of articles. Therefore, we used these 173,259 links between Baidu Baike and English Wikipedia as the seed and finally found 141,754 new cross-lingual links between Baidu Baike and English Wikipedia. Table 6 shows some examples of the discovered cross-lingual links.

Table 6. The examples of the discovered cross-lingual links

Chinese articles (types are in the bracket)	English articles in ranked lists (bold indicates the correct links)
山濑功治 (Person)	**Koji Yamase**, Yokohama Flügels, 1998 Gamba Osaka season, Shinji Tanaka, 1997 J. League
瀛台泣血 (Movie)	**The Last Tempest**, Empress Dowager Cixi, The Empress Dowager, Hundred Days' Reform, The Last Emperor
维容 (Location)	**Vijon**, Salleron, Réunion, Bouzanne, Besanon
查尔斯·泰勒 (Person)	Liberia, **Charles Taylor (Liberian politician)**, Special Court for Sierra Leone, Sierra Leone, Second Liberian Civil War

6 Conclusion

In this paper, we propose an efficient and effective Cross-Lingual Entity Matching approach (CL-EM) to enrich the existing cross-lingual links by learning to rank framework with some language-independent features. We verified our approach on the existing cross-lingual links between Chinese Wikipedia and English Wikipedia by comparing it with other state-of-art approaches. In addition, we also discovered 141,754 new cross-lingual links between Baidu Baike and English Wikipedia, which almost doubles the number of the existing cross-lingual links.

Acknowledgements. This work is supported by the Zhejiang Provincial Natural Science Foundation of China (No. LY17F020015), the Chinese Knowledge Center of Engineering Science and Technology (CKCEST), and the Fundamental Research Funds for the Central Universities (No. 2017FZA5016).

References

1. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2001)
2. Böhm, C., de Melo, G., Naumann, F., Weikum, G.: Linda: distributed web-of-data-scale entity matching. In: CIKM, pp. 2104–2108. ACM (2012)
3. Dai, A.M., Olah, C., Le, Q.V.: Document embedding with paragraph vectors. CoRR abs/1507.07998 (2015)
4. Gabrilovich, E., Markovitch, S.: Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In: IJCAI (2007)
5. Joachims, T.: Optimizing search engines using clickthrough data. In: KDD (2002)
6. Lacoste-Julien, S., Palla, K., Davies, A., Kasneci, G., Graepel, T., Ghahramani, Z.: SIGMa: simple greedy matching for aligning large knowledge bases. In: KDD, pp. 572–580. ACM (2013)
7. Le, Q.V., Mikolov, T.: Distributed representations of sentences and documents. In: ICML (2014)
8. Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P.N., Hellmann, S., Morsey, M., van Kleef, P., Auer, S., Bizer, C.: DBpedia - a large-scale, multilingual knowledge base extracted from Wikipedia. Semantic Web **6**, 167–195 (2015)
9. Lesnikova, T., David, J., Euzenat, J.: Interlinking English and Chinese RDF data sets using machine translation. In: KNOW@LOD (2014)
10. Mahdisoltani, F., Biega, J., Suchanek, F.M.: Yago3: a knowledge base from multilingual Wikipedias. In: CIDR (2015)
11. Navigli, R., Ponzetto, S.P.: BabelNet: the automatic construction, evaluation and application of a wide-coverage multilingual semantic network. Artif. Intell. **193**, 217–250 (2012)
12. Sorg, P., Cimiano, P.: Cross-language information retrieval with explicit semantic analysis. In: CLEF (2008)
13. Sorg, P., Cimiano, P.: Enriching the crosslingual link structure of Wikipedia - a classification-based approach. In: AAAI Workshop on Wikipedia and Artificial Intelligence (2008)
14. Suchanek, F.M., Abiteboul, S., Senellart, P.: Paris: probabilistic alignment of relations, instances, and schema. Proc. VLDB Endow. **5**(3), 157–168 (2011)

15. Vesdapunt, N., Bellare, K., Dalvi, N.: Crowdsourcing algorithms for entity resolution. Proc. VLDB Endow. **7**(12), 1071–1082 (2014)
16. Wang, J., Kraska, T., Franklin, M.J., Feng, J.: Crowder: crowdsourcing entity resolution. Proc. VLDB Endow. **5**(11), 1483–1494 (2012)
17. Wang, J., Li, G., Kraska, T., Franklin, M.J., Feng, J.: Leveraging transitive relations for crowdsourced joins. In: SIGMOD, pp. 229–240. ACM (2013)
18. Wang, Z., Li, J.Z., Wang, Z., Tang, J.: Cross-lingual knowledge linking across wiki knowledge bases. In: WWW (2012)
19. Zwicklbauer, S., Seifert, C., Granitzer, M.: Robust and collective entity disambiguation through semantic embeddings. In: SIGIR (2016)