IJCLA VOL. 3, NO. 2, JUL-DEC 2012, PP. 135–150
RECEIVED 30/11/12 ACCEPTED 10/12/12 FINAL 12/12/12

Entity Linking by Leveraging Extensive Corpus and Semantic Knowledge

YUHANG GUO, BING QIN, TING LIU, AND SHENG LI

Harbin Institute of Technology, China

ABSTRACT

Linking entities in free text to the referent knowledge base entries, namely, entity linking is attractive because it connects unstructured data with structured knowledge. An essential part of this task is the modeling of the entity. Several methods have been proposed to this problem, but they suffer from the sparseness problem. In this paper, we present a new approach to the entity modeling. This approach models an entity by leveraging extensive entity-related corpus to overcome the sparseness, and alleviates the data imbalance between popular and unpopular entities by smoothing. Furthermore, we propose a novel model for the entity linking, which combines contextual relatedness and semantic knowledge. Experimental results on two benchmark data sets show that our proposed approach outperforms the state-of-the-art methods significantly.

KEYWORDS: *Entity Linking, Data Imbalance, Smoothing, Semantic Knowledge*

1 INTRODUCTION

Bridging unstructured text with structured knowledge is widely needed in many natural language processing and data mining tasks. In recent years, as large scale knowledge bases (e.g. Wikipedia[1]) become available, the

[1] http://www.wikipedia.org

entity linking task, which links named entities in free text to the referent knowledge base entries, is attracting more and more attentions.

The major challenge of entity linking is that in natural language a name may refer to different entities in different contexts (i.e. the name ambiguity). In the past, many disambiguation methods have been proposed and gained certain success[1–7]. An essential part of the task is entity modeling. In previous methods, an entity is usually modeled as bag-of-words to measure the contextual similarity between the entity and the surrounding text. In the bag-of-words model, the entity is represented in a term vector of the corresponding entity-description text (e.g. the content of the entity's Wikipedia page). The term here may indicate a word, a named entity or a phrase. However, due to the limited amount of the entity-description text, such model suffers from sparseness problem. Therefore, additional features are incorporated to enhance this model, such as Wikipedia category tags[1], topics[8–12] and neighboring entities[2, 13–18]. However, these features depend on specific knowledge bases[1], need high complexity computation[13] and also suffer from the sparseness problem.

On the other hand, a virtue of the modern knowledge bases (e.g. Wikipedia, DBpedia[19], etc.) is that they contain not only large amount of entities but also massive internal links. In Wikipedia, the number of internal links is over 25 times as the number of articles[2]. These links directly lead a reader to the pages of the entities which are mentioned in the article. Assuming that an entity is related to the text where it is linked, all such texts can be harvested and combined as the training text of this entity. Here we call the combined entity-related training text *entity document*.

However, the above method brings a new problem: the data imbalance. Because popular entities are usually linked by more articles than unpopular ones, the *entity document* size of the popular entities is much bigger than the unpopular entities. This highly skewed distribution will harm system performance. In previous, the training data has to be reduced due to the data imbalance although potentially useful information may be lost [2, 7].

In this paper, we propose an approach to alleviate the data imbalance problem without the data reduction. Our approach is based on language model smoothing. Specifically, we compare two smoothing methods: Jelinek-Mercer smoothing[20] and Dirichlet prior smoothing[21].

[2] http://stats.wikimedia.org/EN/TablesWikipediaEN.htm

The two methods perform similarly in traditional information retrieval [22]. Interestingly, in entity linking the Dirichlet prior smoothing is efficient to the data imbalance problem and outperforms the Jelinek-Mercer smoothing.

Moreover, since the objective of entity linking is to find the referent entity rather than the most contextually related texts, the effect of semantic features should not be underestimated. However, in the basic language model, little semantic information is used. In this paper, we propose a novel probabilistic model which we call *alias model*. This model can not only capture contextual relatedness between the context and entity but also leverage semantic knowledge in the context to distinguish the referent entity from other contextually related entities. Evaluation on two benchmark data sets indicates that our proposed method performs better than the state-of-the-art entity linking significantly.

2 Problem and Approach Overview

Let $\mathcal{E}$ be the set of all entities in the real world. $\mathcal{K} \subseteq \mathcal{E}$ is a knowledge base. Each entity $e \in \mathcal{K}$ has a set of attributes, such as names/aliases, description texts and cross references to other entities, etc.

The entity linking problem is the following: for a name mention m in a given query document d, find the referent entity e in $\mathcal{K}$, if $e \notin \mathcal{K}$ return `NIL`. The query name mention and the query document constitute a query as the input and the referent entity or `NIL` is the expected output.

We evaluate our approach on two data sets: KBP2009 and KBP2010, which are taken from the Knowledge Base Population (KBP) Track[23, 4, 24]. KBP2009 contains 3,904 queries, in which all the query documents are newswire articles. KBP2010 contains 2,250 queries. The query documents of 1,500 queries are newswire articles and the rest 750 are web texts. KBP2009 and KBP2010 share the same knowledge base which is derived from Wikipedia and contains 818,741 entities. In both of the data sets, over a half of the referent entities (2229/3904 and 1230/2250) are absent from the track knowledge base and should be labeled as `NIL`.

The entity linking task can be broken down into two steps: candidate generation and candidate ranking. For the first step, the system selects candidate entities which may be represented in the form of the query name. This step reduces the cost from computing all entities down to a much smaller set of entities. For the second step, the system ranks the candidates and output the top rank candidate. In this paper, we mainly focus on the ranking method and use a simple method to detect the `NIL`

answer: if the candidate set is empty or the top ranked entity is absent from the track knowledge base then return `NIL`.

3 CANDIDATE GENERATION

The goal of the candidate generation is to obtain as many potential entities as possible for the given query name. Wikipedia provides *disambiguation pages* and *redirect pages* from which the candidates can be obtained. However the coverage of this method is not enough for this task because the query name may not be included in the *disambiguation pages* and the *redirect pages*. In this work, we explore the name variations for each entity in Wikipedia and construct a name-entity mapping. The candidates are then generated from this mapping directly.

Given an entity, we extract its names from the following sources in Wikipedia: *title*, *redirect page titles*, *disambiguation page titles*, *bold text* in the first paragraph of the entity, *name field* in the Infobox (e.g. *"name"*, *"birth_name"* or *"nick_name"*), and the *anchor text* of the hyperlinks which link to the entity.

We use the Jun. 20, 2011 version of English Wikipedia dump. In all 140.7 million name-entity pairs which contain 17.3 million names and 3.7 million entities. The name-entity mapping also includes the co-occurrence frequency of the name-entity pairs in Wikipedia. We published this data so that researchers can reproduce our results.

In general, acronym name is more ambiguous than the relevant full name and hence is more difficult to be disambiguated. For example, the acronym *ABC* can be mapped to 79 entities in our name-entity mapping. Meanwhile, the full name *All Basotho Convention* is unambiguous. Fortunately, in some cases the acronyms can be extended to their full forms according to the query document. The following cases are considered:

- The acronym is in a pair of parentheses and the full name is in front of the acronym. (e.g. ... *the newly-formed All Basotho Convention (ABC) is far from certain* ...)
- The full name is in a pair of parentheses and the acronym is in front of the full name. (e.g. ... *at a time when the CCP (Chinese Communist Party) claims* ...)
- The acronym consists of the initial letters of the full name words. (e.g. ... *leaders of Merkel's Christian Democratic Union* ... *CDU* ...)

Given a query name, if it is an acronym, we first attempt to extend its acronym in the query document and then substitute the query name with

the full name. We search the query name in the name-entity mapping and obtain the corresponding candidate entities.

The candidate generation recalls for the non-NIL queries are 91.6% and 94.9% on KBP2009 and KBP2010 data set respectively. Table 1 shows the number of all unique candidates and the average number of candidates per query.

Table 1. Result of the candidate generation on KBP2009 and KBP2010 data set.

Data Set	KBP2009	KBP2010
# of queries	3904	2250
# of non-NIL queries	1675	1020
# of unique candidates	7706	23682
# of candidates/query	22	35
Recall of non-NIL queries	91.6%	95.4%

4 CANDIDATE RANKING

In this section, we present a probabilistic model for the candidate ranking. Next we show the data imbalance between popular and unpopular entities and present how to alleviate the imbalance in the model estimation. Then we propose a novel probabilistic model for entity linking, the *alias model*, which can improve system performance by leveraging semantic knowledge.

4.1 *Probabilistic Model*

In this model, a document is considered as "generated" from a word distribution (e.g. language model). In this sense, the query document d is generated in the following steps: the document author first chooses the entity in mind (the knowledge base), as well as the corresponding name he/she wants to present, and then selects contextual words according to the language model of the entity.

Formally, let e and m denote the referent entity and the mention to be disambiguated. The objective function of entity linking is:

$$e^* = \arg\max_e P(e, m)P(d|e) \tag{1}$$

where $P(e, m)$ is the prior probability of e and m, and $P(d|e)$ is the generative probability of d from the model of e.

Let $f(e, m)$ denote the co-occurrence frequency of entity e and its mention m in the name-entity mapping. The maximum likelihood estimation of the prior probability is

$$P(e, m) = \frac{f(m, e)}{\sum_{e', n'} f(n', e')} \tag{2}$$

where $e' \in \mathcal{K}$, $n' \in N(e')$. $N(e')$ is the set of all names of e'. $P(e, m)$ is the probability of a random observed entity-name pair (e', n') is just the pair (e, m) we concern.

The unigram language model assigns the probability

$$P(d|e) = \prod_{t \in d} P(t|\theta_e) \tag{3}$$

This is the likelihood of the query document d according to the model of entity e. $P(t|\theta_e)$ is the probability of document term t generated by the model of e. Here we assume the terms are sampled from a multinomial distribution. The model parameter θ_e is the multinomial distribution parameter over terms.

The query document is modeled as a bag of terms surrounding the mention m within a window in d. In our approach a term is a name in the name-entity mapping. In our experiment, we extract the terms from the document by using forward maximum matching algorithm which is adopted from word segmentation [25]. We set the window size 50 according to the setting of [26].

Let $C(e)$ denote a bag of terms taken from the training text of e. Let $c(t, C(e))$ be the count of t in $C(e)$. The maximum likelihood estimation of $P(t|\theta_e)$ is

$$P_{ML}(t|\theta_e) = \frac{c(t, C(e))}{|C(e)|} \tag{4}$$

where $|C(e)| = \sum_{t' \in V} c(t', C(e))$ is the length of the training text and V is the set of all names in the name-entity mapping.

4.2 *Data Imbalance*

In previous methods, the entity model is usually trained by using the entity-description text (e.g. the Wikipedia page of the entity). However,

this modeling suffers from sparseness because the lengths of many entity-description texts are not enough to train robust models. In this work, we overcome the sparseness by leveraging extensive corpus (i.e. the *entity document*). The corpus is automatically derived from all the articles which contain a hyperlink leading to the entity to be modeled. However, the *entity document* lengths vary dramatically from popular entities to unpopular entities (e.g. from millions of terms to several terms). Figure 1 shows the *entity document* length distribution on two data sets. Note that the vertical axis is in log scale. From the curves we can see that the distribution approximately obeys Zipf's law[27]. On the both data sets, the longest *entity document* is `United States`, which contains 118 million terms. The average *entity document* length on KBP2009 and KBP2010 are 1.61×10^5 and 1.42×10^5, respectively. And the standard deviations are 1.70×10^6 and 1.20×10^6, respectively. This means that the *entity document* length distribution is highly imbalanced.

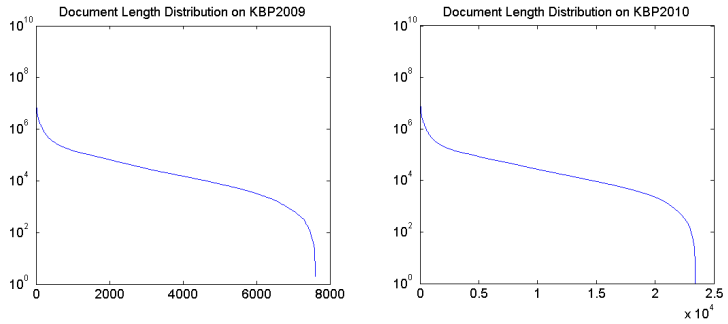

Fig. 1. *Entity document* length distribution over all the candidates on KBP2009 and KBP2010. The horizontal axis is the candidates ranked by the *entity document* length. The vertical axis is the *entity document* length in logarithm scale.

The data imbalance problem is common in knowledge bases because popular entries always have longer description texts and are cited by more articles than unpopular ones. As the growth of the knowledge bases, this information gap will be even larger. Because highly skewed distribution will harm system performance, the training corpus has to be reduced: drop some of the citation articles[2], or set a window around the citation of the entity[7]. Obviously, in this way some useful information about the entity will be lost. In this paper, we propose to employ all the entity-

related text for the training and alleviate the imbalance based on smoothing method.

4.3 *Smoothing*

Because in the maximum likelihood estimation $P_{ML}(t|\theta_e)$, the probability of unseen terms in $C(e)$ is zero, assigning non-zero probability according to some "background knowledge" to the unseen terms (i.e. smoothing) is critical to the accuracy of the model. The "background knowledge" here is the occurrence probability of t in the whole training corpus collection, which is called background model.

$$P(t|\theta_b) = \frac{\sum_{e' \in \mathcal{K}} c(t, C(e'))}{\sum_{e' \in \mathcal{K}} |C(e')|} \tag{5}$$

where θ_b denotes the parameter of the background model.

A direct smoothing is to combine the maximum likelihood estimate and the background model by linear interpolation. This method is also called Jelinek-Mercer smoothing (JM)[20], which is widely used in traditional information retrieval.

$$P(t|\theta_e) = \lambda P_{ML}(t|\theta_e) + (1 - \lambda) P(t|\theta_b) \tag{6}$$

where $\lambda \in (0, 1)$ is a smoothing parameter to control the proportion of the background model.

JM assigns the same background model proportion for each entity. However, if a small λ is assigned to a "long" (*entity document* length) entity, the distribution feature of the entity will be diluted by the background model. On the other hand, if a big λ is assigned to a "short" entity, the estimate of the entity model will be sparse. Since the *entity document* length varies dramatically, it is difficult to find a proper λ to provide good estimates for both "long" and "short" entities.

An alternate smoothing, Dirichlet prior smoothing (DP)[21], adds background model into the conjugate prior of the language model. The smoothed estimate is

$$P(t|\theta_e) = \frac{|C(e)|}{|C(e)| + \mu} P_{ML}(t|\theta_e) + \frac{\mu}{|C(e)| + \mu} P(t|\theta_b) \tag{7}$$

where μ is a smoothing parameter.

Comparing with JM, DP also interpolates maximum likelihood estimate with background model. But the interpolation coefficient in DP is

affected by the length of the *entity document*. For a fixed μ, the model of a "short" entity will be close to the background model, and the model of a "long" entity will be close to the maximum likelihood estimate. Therefore, both of the "short" and the "long" entities can benefit from this estimation at the same time. In the experiment section, we will show that the DP can alleviate the data imbalance and outperforms JM significantly.

4.4 *Alias Model*

Language modeling approaches can capture contextual relatedness between texts. However, a drawback of basic language model is that only the term features are used. In this work, we combine the contextual language model and more discriminative semantic features in a probabilistic framework.

Intuitively, if several different names/aliases or an unambiguous name of a candidate is observed in the document, the confidence on this candidate will increase. How to incorporate the name features into the model is a problem. To this end, we propose a alias model which highlights the name variations of the referent entity in the document.

Let n denote one of the names/aliases of e, we have

$$P(e,m,d) = \sum_{n \in N(e)} P(n,e,m,d) = P(e,m)P(d|e) \sum_{n \in N(e)} P(n|e,d) \tag{8}$$

where $P(e,m)$ and $P(d|e)$ can be estimated as in the basic language model. We approximate the sum factor by

$$\sum_{n \in N(e)} P(n|e,d) = \sum_{n \in N(e) \bigcap d} P(n|e) \tag{9}$$

The maximum likelihood estimation of $P(n|e)$ is

$$P(n|e) = \frac{f(n,e)}{f(e)} = \frac{f(n,e)}{\sum_{n' \in N(e)} f(n',e)} \tag{10}$$

where $f(e)$ is the frequency of entity e in the name-entity mapping.

Table 2 shows an example of how the basic language model can be improved by the name variations in the query document. In this example, the query name mention is *UT* and the referent entity is `University of Tampa`. In the basic language model, the score[3] of `University`

[3] Here the score is $\log(P(e,m,d))$.

of Texas at Austin is much higher than other candidates including University of Tampa because the former entity is more contextual related to the query document. But if we notice that an alias, *Tampa*, of University of Tampa appears in the query document, our confidence on linking *UT* to University of Tampa will be higher. In the alias model the score of University of Tampa increases and is higher than University of Texas at Austin. The name variations such as *Tampa* may appear in the language model of the string University of Tampa too, but the weight of these important features will be diluted by the large size of the *entity document*. In the alias model, such semantic discriminative terms are emphasized separately.

Table 2. An example of the model comparison (using Dirichlet prior smoothing).

Candidates (e)	$f(n,e)$		$f(e)$	$\sum P(n\|e)$	BLM score	AM score
	UT	*Tampa*				
University of Texas at Austin	14	0	6,901	0.0020	–256.44	–262.64
University of Tampa	3	114	449	0.2606	–260.66	–262.00

5 EXPERIMENTS

The evaluation metric is micro-averaged accuracy across the query set, that is, the proportion of the queries which are labeled the correct entity id in knowledge base (or NIL) by the system.

In order to compare with the previous systems, the first evaluation was conducted on KBP2009. We compared our methods with the top three system performances in the track (Siel_093, QUANTA1 and htlcoe1) and four systems reported in [7]:

- The cosine similarity-based method on bag of words features: BoW [2];
- The link similarity based method: TopicIndex[28];
- The improved link based method using machine learning techniques to balance the semantic relatedness, commonness and context quality: Learning2Link[29];

– The entity-mention model proposed by [7]: EMM. EMM is a language model based system with Jelinek-Mercer smoothing but is trained on balanced training data.

The systems that we implemented are trained on extensive training texts, including the basic language model based methods using Jelinek-Mercer smoothing and Dirichlet prior smoothing respectively: BLM-JM, BLM-DP and the alias model using the two smoothing methods: AM-JM, AM-DP.

The system performances on KBP data sets are shown in Table 3, where the three columns represent the system performance on: All queries (All), in-knowledge-base-answer queries (inKB) and NIL-answer queries (NIL) respectively. The results of BLM-JM, AM-JM, BLM-DP and AM-DP are optimal over the smoothing parameters which we have searched. AM-DP* on KBP2009 is the empirical optimal AM-DP result tuned on KBP2010 and AM-DP* on KBP2010 is tuned on KBP2009. The optimal values of λ and μ are shown in Table 4.

Table 3. Results on the KBP data set.

(a) KBP2009

	All	inKB	NIL
Siel_093	0.82	0.77	0.86
QUANTA1	0.80	0.77	0.83
hltcoe1	0.79	0.71	0.87
BoW	0.72	0.77	0.65
TopicIndex	0.80	0.65	0.91
Learning2Link	0.83	0.73	0.90
EMM	0.86	0.79	0.90
BLM-JM	0.84	0.75	0.91
AM-JM	0.86	0.77	0.92
BLM-DP	0.87	**0.81**	0.91
AM-DP	**0.88**	**0.81**	**0.93**
AM-DP*	**0.88**	**0.81**	**0.93**

(b) KBP2010

	All	inKB	NIL
LCC	0.86	0.79	**0.91**
Siel	0.82	0.72	0.90
CMCRC	0.82	0.74	0.89
KL	0.85	0.81	0.87
BLM-JM	0.84	0.79	0.89
AM-JM	0.85	0.79	0.90
BLM-DP	**0.88**	**0.84**	0.90
AM-DP	**0.88**	**0.84**	**0.91**
AM-DP*	**0.88**	**0.84**	**0.91**

As can be seen from Table 3(a), on KBP2009, our proposed method (i.e. AM-DP*) outperforms the best ranking system in the KBP track 2009 by 6% improvement. Compared with the BoW, TopicIndex, and Learning2Link baselines, our proposed method gets 16%, 8%, 5% im-

provements respectively. Our method also performs significantly better than the state-of-the-art method: EMM (under Z-test with $p < 0.01$).

Table 4. Optimal parameters of Jelinek-Mercer (JM) smoothing and Dirichlet prior (DP) smoothing on KBP2009 and KBP2010 data sets.

Smoothing	JM(λ)		DP($\mu \times 10^6$)	
Data Set	2009	2010	2009	2010
BLM	0.8	0.9	3.7	4.0
AM	0.75	0.7	3.7	4.0

Table 3(b) shows the system performances on KBP2010. We compared our method with the top three systems in the track (LCC, Siel and CMCRC) and a recently proposed KL-divergence based method: KL[30]. Our proposed method also outperforms the best system significantly ($p < 0.05$).

On the both data sets, the alias model performs better than the basic language model. DP outperforms JM significantly ($p < 0.01$) in the basic language model (up to 4%). AM-DP outperforms BLM-JM significantly by 4% ($p < 0.01$.).

The improvement of the alias model on KBP2009 is more than that on KBP2010. This is because the KBP2009 query documents are all news articles which are rich in names whereas the KBP2010 query documents consist of news and web text. Therefore, the number of effective aliases in each document of KBP2009 are more than KBP2010 on average. This indicates that the alias model performs better on the query documents which are rich in names.

6 CONCLUSIONS

In this paper, we propose to use extensive entity-related corpus to overcome the sparseness problem in the entity modeling of entity linking. Due to the highly skewed distribution of the entities, the training data for the entity modeling is highly imbalanced. We investigate language modeling based approaches and find that Dirichlet prior smoothing performs better than Jelinek-Mercer smoothing because it can leverage the length of entity entity's training text. The property of Dirichlet prior smoothing makes it suitable for the data imbalance scenario. We further combine the

contextual language modeling and name variation feature in a probabilistic framework and propose an alias model. Experimental results on two standard test sets show that the Dirichlet prior smoothing performs better than Jelinek-Mercer smoothing and our proposed model outperforms the state-of-the-art performance significantly.

ACKNOWLEDGEMENTS This work was supported by National Natural Science Foundation of China (NSFC) via grant 61273321, 61073126, 61133012 and the National 863 Leading Technology Research Project via grant 2012AA011102.

REFERENCES

1. Bunescu, R.C., Pasca, M.: Using encyclopedic knowledge for named entity disambiguation. In: EACL, The Association for Computer Linguistics (2006)
2. Cucerzan, S.: Large-scale named entity disambiguation based on Wikipedia data. In: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), Prague, Czech Republic, Association for Computational Linguistics (June 2007) 708–716
3. Mihalcea, R., Csomai, A.: Wikify!: linking documents to encyclopedic knowledge. In: CIKM '07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management, New York, NY, USA, ACM (2007) 233–242
4. Dredze, M., McNamee, P., Rao, D., Gerber, A., Finin, T.: Entity disambiguation for knowledge base population. In: Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010), Beijing, China, Coling 2010 Organizing Committee (August 2010) 277–285
5. Zheng, Z., Li, F., Huang, M., Zhu, X.: Learning to link entities with knowledge base. In: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Los Angeles, California, Association for Computational Linguistics (June 2010) 483–491
6. Zhang, W., Sim, Y.C., Su, J., Tan, C.L.: Entity linking with effective acronym expansion, instance selection, and topic modeling. In Walsh, T., ed.: IJCAI 2011, Chiang Mai, Thailand (November 2011) 1909–1914
7. Han, X., Sun, L.: A generative entity-mention model for linking entities with knowledge base. In: 49th Annual Meeting of the Association for Computational Linguistics: Human Language Techologies, Portland, Oregon, USA, Association for Computational Linguistics (June 2011) 945–954
8. Cucerzan, S.: TAC entity linking by performing full-document entity extraction and disambiguation. In: Text Analysis Conference 2011. (2011)

9. Kataria, S.S., Kumar, K.S., Rastogi, R.R., Sen, P., Sengamedu, S.H.: Entity disambiguation with hierarchical topic models. In: 17th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '11, New York, NY, USA, ACM (2011) 1037–1045
10. Zhang, W., Su, J., Tan, C.L.: A Wikipedia-LDA model for entity linking with batch size changing instance selection. In Walsh, T., ed.: IJCAI 2011, Chiang Mai, Thailand (November 2011) 562–570
11. Sen, P.: Collective context-aware topic models for entity disambiguation. In: 21st international conference on World Wide Web, WWW '12, New York, NY, USA (2012) 729–738
12. Han, X., Sun, L.: An entity-topic model for entity linking. In: 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, Jeju Island, Korea, Association for Computational Linguistics (July 2012) 105–115
13. Kulkarni, S., Singh, A., Ramakrishnan, G., Chakrabarti, S.: Collective annotation of Wikipedia entities in web text. In: 15th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '09, New York, NY, USA, ACM (2009) 457–466
14. Han, X., Sun, L., Zhao, J.: Collective entity linking in web text: a graph-based method. In: 34th international ACM SIGIR conference on Research and development in Information, SIGIR '11, New York, NY, USA (2011) 765–774
15. Hoffart, J., Yosef, M.A., Bordino, I., Fürstenau, H., Pinkal, M., Spaniol, M., Taneva, B., Thater, S., Weikum, G.: Robust disambiguation of named entities in text. In: 2011 Conference on Empirical Methods in Natural Language Processing, Edinburgh, Scotland, UK. (July 2011) 782–792
16. He, J., de Rijke, M., Sevenster, M., van Ommering, R., Qian, Y.: Generating links to background knowledge: a case study using narrative radiology reports. In: 20th ACM international conference on Information and knowledge management, CIKM '11, New York, NY, USA (2011) 1867–1876
17. Ratinov, L., Roth, D., Downey, D., Anderson, M.: Local and global algorithms for disambiguation to Wikipedia. In: 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Portland, Oregon, USA (June 2011) 1375–1384
18. Shen, W., Wang, J., Luo, P., Wang, M.: Linden: linking named entities with knowledge base via semantic knowledge. In: 21st international conference on World Wide Web, WWW '12, New York, NY, USA (2012) 449–458
19. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: Dbpedia: a nucleus for a web of open data. In: 6th international conference The semantic web and 2nd Asian conference on Asian semantic web, ISWC'07/ASWC'07, Berlin, Heidelberg, Springer-Verlag (2007) 722–735
20. Jelinek, F., Mercer, R.L.: Interpolated estimation of Markov source parameters from sparse data. In: Workshop on Pattern Recognition in Practice, Amsterdam, The Netherlands, North-Holland (May 1980) 381–397

21. MacKay, D.J.C., Peto, L.C.B.: A hierarchical dirichlet language model. Natural Language Engineering **1** (1995) 289–308
22. Zhai, C., Lafferty, J.: A study of smoothing methods for language models applied to information retrieval. ACM Trans. Inf. Syst. **22** (April 2004) 179–214
23. McNamee, P., Dang, H.: Overview of the tac 2009 knowledge base population track. `http://www.nist.gov/tac/publications/2009/presentations/TAC2009_KBP_overview.pdf` (2009)
24. Ji, H., Grishman, R.: Knowledge base population: Successful approaches and challenges. In: 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Portland, Oregon, USA, Association for Computational Linguistics (June 2011) 1148–1158
25. Guo, J.: Critical tokenization and its properties. Comput. Linguist. **23** (December 1997) 569–596
26. Pedersen, T., Purandare, A., Kulkarni, A.: Name discrimination by clustering similar contexts. In Gelbukh, A., ed.: CICLing 2005, Lecture Notes in Computer Science. Volume 3406., Springer (2005) 226–237
27. Zipf, G.K.: Human Behaviour and the Principle of Least Effort: An Introduction to Human Ecology. Addison-Wesley (1949)
28. Medelyan, O., Witten, I.H., Milne, D.: Topic indexing with Wikipedia. In: AAAI WikiAI workshop. Volume 1., AAAI Press (2008) 19–24
29. Milne, D., Witten, I.H.: Learning to link with wikipedia. In: CIKM '08: Proceeding of the 17th ACM conference on Information and knowledge management, New York, NY, USA, ACM (2008) 509–518
30. Gottipati, S., Jiang, J.: Linking entities to a knowledge base with query expansion. In: 2011 Conference on Empirical Methods in Natural Language Processing, Edinburgh, Scotland, UK., Association for Computational Linguistics (July 2011) 804–813

YUHANG GUO
RESEARCH CENTER FOR SOCIAL COMPUTING AND INFORMATION RETRIEVAL,
HARBIN INSTITUTE OF TECHNOLOGY,
CHINA
E-MAIL: <YHGUO@IR.HIT.EDU.CN>

BING QIN
RESEARCH CENTER FOR SOCIAL COMPUTING AND INFORMATION RETRIEVAL,
HARBIN INSTITUTE OF TECHNOLOGY,
CHINA
E-MAIL: <BQIN@IR.HIT.EDU.CN>

TING LIU
(CORRESPONDING AUTHOR)
RESEARCH CENTER FOR SOCIAL COMPUTING AND INFORMATION RETRIEVAL,
HARBIN INSTITUTE OF TECHNOLOGY,
CHINA
E-MAIL: <TLIU@IR.HIT.EDU.CN>

SHENG LI
RESEARCH CENTER FOR SOCIAL COMPUTING AND INFORMATION RETRIEVAL,
HARBIN INSTITUTE OF TECHNOLOGY,
CHINA
E-MAIL: <SLI@IR.HIT.EDU.CN>