

A Novel Entity Type Filtering Model for Related Entity Finding

J. Zhang, Y. Qu, S. Tian

Junsan Zhang

1.College of Computer & Communication Engineering,
China University of Petroleum,
Qingdao 266580, P.R.China
2.School of Computer and Information Technology,
Beijing Jiaotong University,
Beijing 100044, P.R.China
zhangjunsan@upc.edu.cn

Youli Qu*, Shengfeng Tian

School of Computer and Information Technology,
Beijing Jiaotong University,
Beijing 100044,P.R. China
*Corresponding author: ylqu@bjtu.edu.cn
sftian@bjtu.edu.cn

Abstract: Entity is an important information carrier in Web pages. Searchers often want a ranked list of relevant entities directly rather a list of documents. So the research of related entity finding (REF) is a meaningful work. In this paper we investigate the most important task of REF: Entity Ranking. To address the issue of wrong entity type in entity ranking: some retrieved entities don't belong to the target entity type. We propose a novel entity type filtering model in which the target types are composed of the originally assigned type and the new type which is automatically acquired from the topic's narrative to filter wrong-type entities. For the query, we propose a method to process the original narrative to acquire a new query which is composed of noun and verb phrases. The results of experiments show our novel type filtering model gets a better result than the traditional filtering model at whatever precision and recall. Also the experiment shows the method that we acquire a new query is feasible.

Keywords: related entity finding, entity, entity ranking, type filtering.

1 Introduction

Along with the rapid development of internet, the number of web pages becomes more and more, so mass information is being produced now. Search engine became an important tool to query information from web in people's life. If a user is looking for entities, which have a specific relationship to some entity, he has to scan the documents retrieved by a Search Engine system to look for entities. For instance, when searchers submit a query *Michael's teammates while he was racing in formula 1* [1], searchers want some related entities actually. Related entity finding task can meet searchers requirements. According to the definition of TREC2009 Entity track, related entity finding(REF): given a source entity, a relation and a target type, identify homepages of target entities that enjoy the specified relation with the source entity and that satisfy the target type constrain [1]. REF provides a new way of information searching through entities. Entity ranking is an important issue of REF .Two elements can affect the result of entity ranking, target entity type and entity relation between source entity and target entity. In this paper we focus on the effect of target entity type to entity ranking. Because wrong type entities pollute the result of entity ranking, we try to filter the entities of wrong type. However the common type

type filtering method is too coarse to filter wrong entities exactly. Therefore, we propose a novel type filtering model to filter wrong entities. We utilize the Wikipedia category information as the source of entities types. Also we observe carefully the effect of experiment and we see using the novel type filtering model can get a better result at both recall and precision . To the issue of extracting query , we propose an approach in which parsing the narrative’s syntactic structure and rewriting the query. The paper is organized as follows: Section 2 provides an overview of related work, Section 3 gives a description of the basic architecture of REF, Section 4 gives a detailed description of the proposed method, in Section 5 we utilize data set to implement our proposed method and analyze the experimental results, in Section 6 we draw the conclusion and propose our works in the future.

2 Related Work

The entity retrieval originate natural language processing, specifically IE (information extraction). Finding all entities for a certain class is the target of IE, i.e., extracting entities based patterns or learned from examples or created manually [2]. QA (question answering) is the intersection of natural language processing and IR which combines IE and IR. It looks like the REF, yet it differs from REF: (i) an entity is not always been contained in QA query list [3] (ii) REF task add a special relation between target entities and source entity [1]. As an important issue of REF task, entity ranking begin with ranking a specific type entities, e.g. persons in expert search [4]. The task of expert finding is finding experts either by modeling an expert’s knowledge by its associated documents or collecting topic related documents first and then modeling experts [5]. Now it develops to rank more general types, e.g. persons, products, locations, organizations etc. The goal of entity ranking is retrieving entities as answers to a query. It is primarily focused on returning a ranked list of relevant entities [6]. What our concern are precision and recall. Type filtering can demote the wrong type entities and improve recall and precision. The novelty of our approach is we use co-occurrence model which widely used to estimate the strength of association between terms to estimate the associations between source entity and target entities and using a novel type filtering model filters wrong entities. We apply Wikipedia category information to as the source of entities types. Also we carefully analyze the effect of using the novel filtering model and the traditional filtering model for entity ranking . TREC has run an entity ranking track in 2009 aiming at performing entity-oriented search task on the web [1]. The definition of entity track is: given a source entity, a relation and a target type, find the relevant entities. It makes use of 20 topics, finds three types entities (persons, products, organizations). A query topic is defined as follows [1]:

```
< query >
< num > 1 < /num >
< entity_name > Blackberry < /entity_name >
< entity_url >clueweb09-en0004-50-39593
< /entity_url >
< target_entity > organization < /target_entity >
< narrative >Carriers that Blackberry makes phones for.
< /narrative >
< /query >
```

A general approach of REF task is: (i) collecting text snippets from relevant documents (ii) obtaining entities by performing named entity recognition (iii) ranking relevant entities (iv) finding homepage [1]. Researchers propose several approaches to perform the REF task. Some researchers use different language modeling approaches where the entity model is constructed from text snippets and relation is utilized ad a query [7], [8]. Y.Wu et al. [9] develop an effective

approach to rank entities via measuring the "similarities" between supporting snippets of entities and input query. Y.Fang et al. [10] propose a hierarchical relevance retrieval model for entity ranking. Three levels of relevance are examined which are document, passage and entity, respectively. R. Kaptein et al. [11] propose an approach using Wikipedia as a pivot for finding entities on the web, reducing the hard web entity ranking problem to easier problem of Wikipedia entity ranking.

3 The Basic Architecture of REF

The basic architecture of related entity finding is shown in Figure 1. The REF task can be divided into three main parts: (i) relevant documents retrieving (ii) candidate entities extracting and entity ranking (iii) homepage finding. We will describe the three parts in the following paragraph.

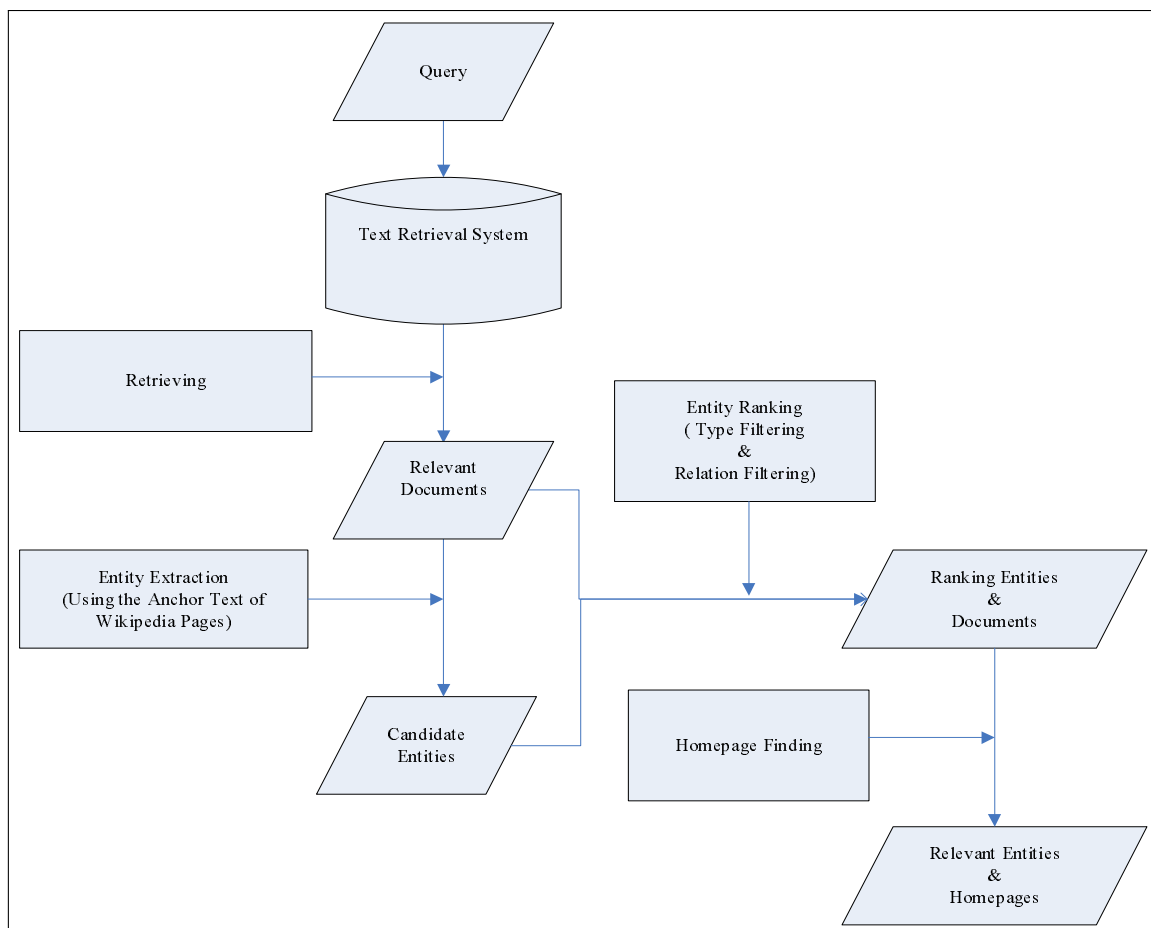


Figure 1: The Basic Architecture of REF

(1) Relevant documents retrieving. Retrieving relevant documents is the basic component of REF task. The first step is using corpus (here, we use the ClueWeb09 Category B as the documents repository) to build a full text retrieval system. Because our computing resource is limited, we make use of "The Lemur Project" [12] which provides an online service of ClueWeb09 Category B as our source. Secondly, we send a query to the retrieval architecture and preliminary generate some candidate answers. For the selection of query, we chose the noun phrases and verb phrases as the query. We are interested in noun groups and verb groups because the

noun groups often qualify the target entity and source entity in more detail and can be seen as a kind of a selection criterion. Through extracting the part-of-speech elements from the narrative, we get the noun phrases and verb phrases of each topic. For example, the topic19's (Entity Track09) narrative is "Companies that John Hennessy serves on the board of". After the parsing the syntactic structure, the query is obtained: "Companies John Hennessy serves serves". For detecting the feasibility of acquiring query through this approach, we respectively utilize the "pure narrative" and the "noun and verb phrases" as the query to retrieve documents in experimental part.

(2) Entity Ranking. Entity ranking is the focus in this paper. After generating the relevant documents, the traditional following step is named entity recognition (NER), yet NER is not our emphasis in this paper. We handle NER by considering only anchor texts as entity occurrences in Wikipedia pages [13], [14]. When we get some candidate entities, we hope to find the most relevant entities. So we need to rank the candidate entities. There are two factors will effect the entity ranking: entity type and entity relation. The wrong type entities and the entities which do not conform to the relation between source entity and target entity will pollute the ranking result. In this paper, We only focus on the issue of "filtering wrong type entities".

(3) Homepage Finding. An entity is uniquely identified by its homepage, according to the definition of REF. Three homepages and a Wikipedia page at most can be returned for each entity in 2009 Entity Track. Homepage finding can be seen a document retrieval problem which employs a standard language modeling [15] to ranks homepages according to the query likelihood: $p(q = e/d)$, using the entity's name as a query. This issue is also not the emphasis in this paper.

4 Entity Ranking Model

According to the definition of REF, given a $Q(E^s, T, R)$, return a ranked list of relevant entities. In the paper, we use E^s to indicate the source entity, E^t indicate the target entity, T indicate the target type, R indicate a relation between E^s and E^t . Using the conditional probability formula $P(E^t|Q)$ estimates REF task. Due to the condition of $P(E^t|Q)$ is complex and difficult to estimating. Next we rewrite $P(E^t|Q)$ to:

$$P(E^t|Q) = \frac{P(E^t, Q)}{P(Q)} \quad (1)$$

Considering the denominator $P(Q)$ does not influence the ranking of entities, we derive the ranking formula as follows:

$$P(E^t, Q) = P(Q|E^t) \cdot P(E^t) \quad (2)$$

$$= P(E^s, T, R|E^t) \cdot P(E^t) \propto P(E^s, R|E^t) \cdot P(T|E^t) \cdot P(E^t) \quad (3)$$

$$= P(E^s, R, E^t) \cdot P(T|E^t) = P(R|E^s, E^t) \cdot P(E^s, E^t) \cdot P(T|E^t) \quad (4)$$

$$= P(R|E^s, E^t) \cdot P(E^t|E^s) \cdot P(E^s) \cdot P(T|E^t) \quad (5)$$

$$= P(R|E^s, E^t) \cdot P(E^t|E^s) \cdot P(T|E^t) \quad (6)$$

We assume that type T is independent of source entity and relation R in (3). Assuming $P(E^s)$ is a uniform value in (5), we drop it. Now the ranking task is converted to three conditional probability question: $P(R|E^s, E^t)$, $P(E^t|E^s)$, $P(T|E^t)$. In this paper, our goal is to address the issue of wrong type polluting entity ranking. So we only discuss $P(E^t|E^s) \cdot P(T|E^t)$ in this paper.

(1) Co-occurrence model

We see $P(E^t|E^s)$ as a co-occurrence issue expresses the association between source entity E^s and target entity E^t . We use a formula to estimate $P(E^t|E^s)$ as flows:

$$P(E^t|E^s) = \frac{Co(E^t, E^s)}{\sum_{E^{t'}} Co(E^{t'}, E^s)} \quad (7)$$

$E^{t'}$ indicates an entity co-occurrence with source entity E^s in documents. We use two approach to estimate $C(E^t, E^s)$: (1) maximum likelihood estimate, (2) χ^2 hypothesis test [13], [16].

Maximum likelihood estimate(MLE):

$$C_{MLE}(E^t, E^s) = C(E^t, E^s)|C(E^s) \quad (8)$$

Where $C(E^t, E^s)$ indicates the number of documents in which E^t and E^s co-occurrence, $C(E^s)$ indicates the number of documents in which $C(E^s)$ occurrence.

χ^2 hypothesis test:

$$Co_{\chi^2}(E^t, E^s) = \frac{N \cdot (C(E^t, E^s) \cdot C(\bar{E}^t, \bar{E}^s) - C(E^t, \bar{E}^s) \cdot C(\bar{E}^t, E^s))^2}{C(E^s) \cdot C(E^t) \cdot (N - C(E^t)) \cdot (N - C(E^s))} \quad (9)$$

Where \bar{E}^t , \bar{E}^s indicate the E^t and E^s don't appears respectively, and N indicates the total number of documents. For example, $C(\bar{E}^t, \bar{E}^s)$ expresses the number of documents in which both E^t and E^s don't appear.

(2) Entity type filtering model

The co-occurrence model preliminary ranks entities. But it can not resolve the problem of wrong type entities pollute the ranking result. To address the issue of wrong type entities will pollute the ranking result. Traditional type filtering model deal with $P(T|E^t)$: the relation between target entity type and candidate entity type as flows:

$$P(T|E^t) = \begin{cases} 1 & \text{if } C(E^t) \cap C(T) \neq \phi \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

Here, the $C(T)$ indicates the expected target entity type and the $C(E^t)$ indicates the type of candidate entity. The former is previously defined, although the latter is acquired via the Wikipedia category information of candidate entity. If they have an intersection we think the probability is 1, otherwise the probability is 0.

Although utilizing traditional entity filtering model can filter some wrong type entities, it is not enough accurate sometimes. According to the definition of REF, the types of target entities are divided into several types which are too wide to a certain extent. Such as, for entity track 2009, there are only three types of target entities which are assigned to 20 topics: person, organization and product. Yet, we see the exact target type of each topic should be different through the observation of topics' narratives. For example, there are two topics which have same target type (person). But they have completely different narratives: "Authors awarded an Anthony Award at Bouchercon in 2007", "Chefs with a show on the Food Network". From the narratives, the exact type which the former want is Authors but the latter want is Chefs. Certainly, authors and chefs are both persons, yet they are also two different kinds of persons. So if we can refine the target type according to the topic's narrative, it may filter wrong type entity more accurately. We propose a novel entity filtering model to estimate $P(E^t|E^s)$ as flows:

$$Score(T) = Score(T_t) + Score(T_n) = \frac{F_{c,t}}{F_t} + \frac{F_{c,n}}{F_n} \quad (11)$$

Where: $Score(T)$ - score of type that a candidate entity get on the whole; $Score(T_n)$ - score of type that a candidate entity get through the type acquired from topic's narrative; $Score(T_t)$ - score of type that a candidate entity get through the topic's assigned target type; $F_{c,t}$ - number of category features that a candidate entity type and the topic's assigned target entity type have in common; F_t - number of category features that the topic's assigned target entity type has; $F_{c,n}$ - number of category features that the candidate entity type and the type acquired from topic's narrative have in common; F_n - number of category features that acquired from topic's narrative.

In order to calculate the $Score(T_t)$, we first get $Cat(t)$ - category information of target entity type that a topic is assigned and its sub-categories (one level down) and $Cat(c)$ - category information of a candidate entity. Then we can acquire $F_{c,t}$ and F_t through the category information of themselves. In this paper, we make use of Wikipedia category information as the criterion of judgment. However, The Wikipedia category structure is not a strict hierarchy and the category assignments are imperfect [17]. So we must process the category information further. The number of features that $Cat(t)$ has is too much to use directly. We find many categories of $Cat(t)$ often have a common structure (starting with "something" and ending with "something"). We select the top five categories or structures that appear in $Cat(t)$ as the selected features. For example, the selected features of a assigned target entity type - "Person" are:

- 'Living People'
- Ending with 'births'
- Ending with 'deaths'
- Starting with 'People'
- Ending with 'People'

For $Score(T_n)$, we first get $Cat(n)$ - the category information of target entity type that we get from the topic's narrative and its sub-categories and $Cat(c)$ - the category information of a candidate entity. Then we can acquire $F_{c,n}$ and F_n through the category information of themselves. To get $Cat(n)$, the category names in the narrative must extract first. We process the topics' narratives using Brill's Part-of-Speech [18] tagger and a Noun-Phrase chunker [19]. The first noun phrase in the narrative is the category that we want. For example, the narrative "Chefs with a show on the Food Network" is processed as follows:

```
(ROOT
(NP
(NP (NNS Chefs))
(PP (IN with)
(NP (DT a) (NN show))))
(PP (IN on)
(NP (DT the) (NNP Food) (NNP Network))))
```

The noun Chefs is the type that is extracted from the narrative. Then we utilize the Wikipedia category information to get $Cat(n)$. Next, the category features of $Cat(n)$ is selected and the process is same as that is described in the previous paragraph. For example the selected features of "Chefs" are:

- 'Chefs'
- Ending with 'Chefs'
- Ending with 'Characters'
- Ending with 'births'
- 'Living People'

5 Experiment

5.1 Dataset and Evaluation Measures

In this paper, we use ClueWeb09 Category B subset as our corpus, including about 50 million documents. TREC 2009 Entity Track has three kind of basic types which are Person, Product and Organization [1]. We restrict the scope of entities only in Wikipedia pages. So we drop 5 topics in which no Wikipedia pages are retrieved and 15 topics are left. The Wikipedia is an excellent structured resource of entities. The title of the page is the name of the entity, the content of the page is the representation of the entity and each Wikipedia page is assigned to a number of categories. In our experiment, we get about 100 thousand relevant entities totally. After culling duplicated entities we get about 70 thousand relevant entities. We also make use of the DBpedia category information to get each entity's category information. DBpedia is a project aiming to extract structured content from the information created as part of the Wikipedia project. This structured information is then made available on the World Wide Web. DBpedia allows users to query relationships and properties associated with Wikipedia resources, including links to other related datasets. We use precision and recall as our estimation measures. Using $P@10$ to express precision and $R@N$ express recall where N taken to be 100, 2000.

In order to evaluate the effects that the different model produces for entity ranking and the feasibility that we acquire query from the original topic narrative. We divide the experiment into five steps: (1) Using the original topic narrative as the query retrieves relevant documents. (2) Using the noun and verb phrases which is chosen from the original topic narrative as the query retrieves relevant documents. (3) We only make use of pure co-occurrence model to rank candidate entities. (4) We make use of traditional entity type filtering method to rank candidate entities based the result of using pure co-occurrence model. (5) Utilizing our novel entity type filtering model to rank candidate entities based the result of using pure co-occurrence model.

5.2 Experimental Result

The precision and recall are our estimation measures. Using $P@10$ to express precision of top 10 entities retrieved and $R@N$ express recall where N taken to be 100, 2000. We utilize chart and table to describe the experimental data, showing the effects of different methods rank entities.

(1) For detecting the feasibility that using noun and verb phrases which are chosen from the narrative as the query, we make use of the narrative and the chosen phrases as the query to retrieve candidate entities respectively. The number of right entities which are retrieved from the documents are shown in Figure 2.

(2) We make use of pure co-occurrence (MLE), traditional type filtering model and our proposed novel type filtering model (based the MLE) to estimate the effect of entity ranking respectively. The result ($P@10$) is shown in Figure 3.

(3) We make use of pure co-occurrence (χ^2), traditional type filtering model and our proposed novel type filtering model (based the χ^2) to estimate the effect of entity ranking respectively. The result ($P@10$) is shown in Figure 4.

(4) We take topic14 as an example and list the top ten entities in Table 1. In order to observe the variation of the top ten ranked entities using different methods, more intuitively.

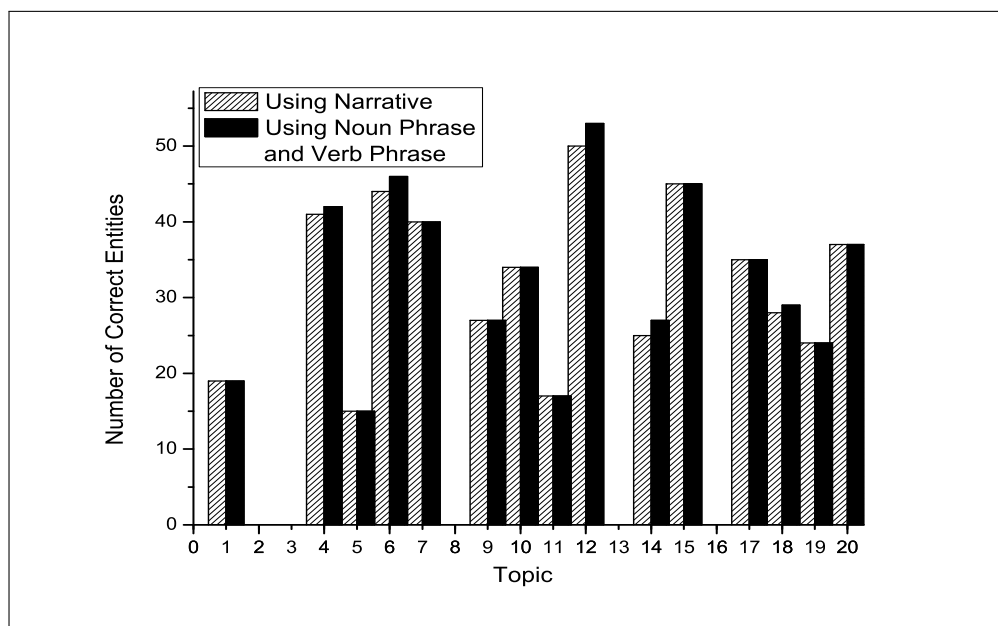


Figure 2: Number of correct entities using different query

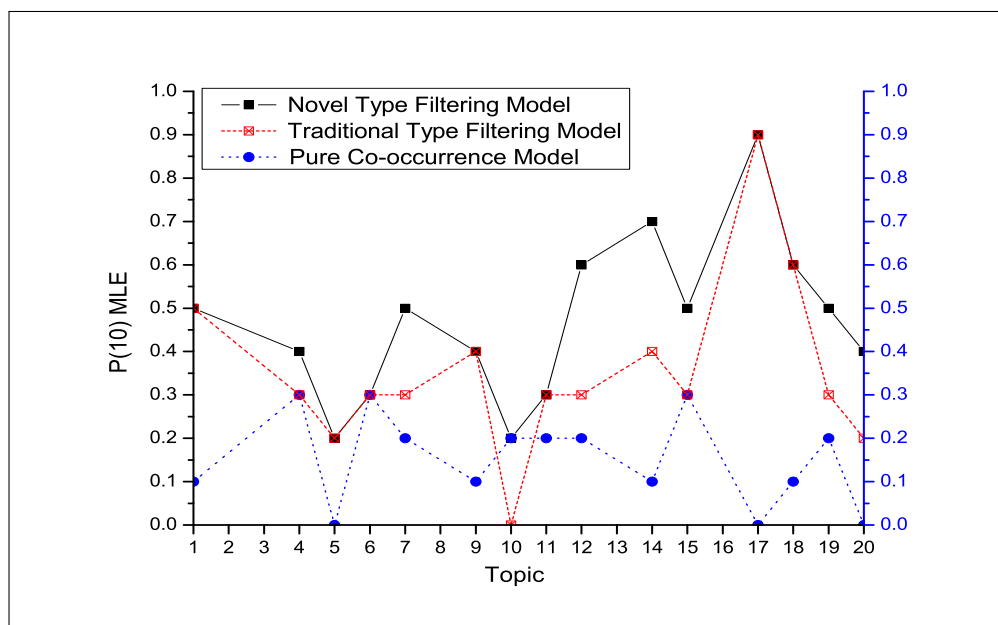


Figure 3: $P@10$: using MLE co-occurrence model, traditional type filtering model and novel type filtering model filters entities respectively.

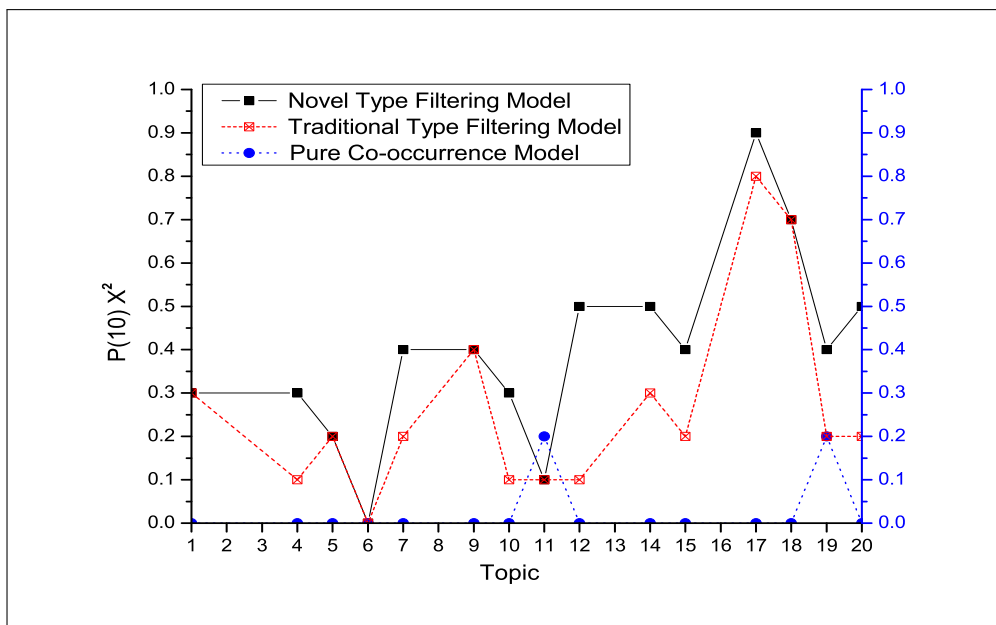


Figure 4: $P@10$: using χ^2 co-occurrence model, traditional type filtering model and novel type filtering model filters entities respectively.

(5) In order to evaluate the experiment effect in the whole. We list the results (the average of precision and recall of all topics) of using different models which are shown in Table 2.

5.3 Analysis

Figure 2 shows the number of correct entities which are retrieved from the documents using original narrative and phrases as queries respectively. We see using noun and verb phrases as queries retrieves more correct entities (increased by 2.07%) than using original narratives as queries. It certifies the approach that we extract noun and verb phrases as query is feasible.

Figure 3 shows the type filtering (based on MLE) whatever using traditional type filtering model or using our novel type filtering model causes a better result than using pure MLE co-occurrence model in $P@10$. The Figure 4 show the type filtering (based χ^2) whatever using traditional type filtering model or using our novel type filtering model causes a better result than using pure co-occurrence model in $P@10$. Through the Figures we find that utilizing type filtering can improve the effects of $P@10$. And our proposed novel type filtering model can filter wrong type entities better than traditional type filtering model. Table 1 demonstrates intuitively the variation of topic14's top ten ranked entities. Our novel type filtering model effectively removes the wrong type entities from the ranking.

As to recall, Table 2 shows the $R@100$ and $R@2000$ of using pure co-occurrence model, traditional type filtering model and using our novel type filtering model respectively. We see using the novel type filtering model (based on MLE) increases by 27.35% than using traditional type filtering model (based on MLE) and increases by 54.43% than using pure MLE co-occurrence model in $R@100$. Also we see using the novel type filtering model (based on χ^2) increases by 50% than using traditional type filtering model (based on χ^2) in $R@100$. The result illustrates than our novel type filtering model can filter wrong type entities than traditional type filtering model in $R@100$.

| Pure Co-occurrence Model | | Traditional Type Filtering Model | | Novel Type Filtering Model | |
|--------------------------|-------------|----------------------------------|------------------------|----------------------------|--------------------------|
| MLE | χ^2 | MLE | χ^2 | MLE | χ^2 |
| Bouchercon | Music | Sue Grafton | Sherlock Holmes | Sue Grafton | Lawrence Block |
| Book | Publisher | Agatha Christie | Sue Grafton | Ross Macdonald | Sue Grafton |
| Navigation | Husband | Edgar Allan Poe | Edgar Allan Poe | Edgar Allan Poe | Bill Pronzini |
| History | Television | Sherlock Holmes | Danny Boyle | George Washington | Lee Child |
| US | Crime | Robert Crais | Hercule Poirot | Agatha Christie | Marcia Muller |
| Son | Books | George Washington | George Washington | Robert Crais | Ross Macdonald |
| Organization | Performance | Hercule Poirot | Agatha Christie | Dennis Lehane | George Washington |
| Ant | Homicide | Stieg Larsson | Laurence Olivier | Michael Connelly | Max Allan Collins |
| Cher | Detective | Marcia Muller | Lawrence Block | Lee Child | Laura Lippman |
| Pub | Big | Ross Macdonald | Bill Pronzini | Val McDermid | Val McDermid |

Table 1: The top ten ranked entities of topic14. The correct entities are indicated in bold.

| | | $P@10$ | $R@100$ | $R@2000$ |
|----------------------------------|----------|--------|---------|----------|
| Pure Co-occurrence Model | MLE | 0.1533 | 0.1646 | 0.4287 |
| | χ^2 | 0.0267 | 0.0438 | 0.3968 |
| Traditional Type Filtering Model | | $P@10$ | $R@100$ | $R@2000$ |
| | MLE | 0.3533 | 0.1996 | 0.2674 |
| Novel Type Filtering Model | χ^2 | 0.2600 | 0.1492 | 0.2674 |
| | | $P@10$ | $R@100$ | $R@2000$ |
| Novel Type Filtering Model | MLE | 0.4667 | 0.2542 | 0.3632 |
| | χ^2 | 0.3933 | 0.2238 | 0.3627 |

Table 2: Recall and Precision of using different type filtering model filters entities respectively.

However, the data illustrates some different things in $R@2000$. We see using novel type filtering model (based on MLE) reduces by 15.28% than using pure MLE co-occurrence model and using traditional type filtering model (based MLE) reduces by 38.25% than using pure MLE co-occurrence model. And using novel type filtering model (based on χ^2) reduces by 8.47% than using pure χ^2 co-occurrence model and using traditional type filtering model (based on χ^2) reduces by 32.61% than using pure χ^2 co-occurrence model. The result illustrates that the type filtering model is not accurate enough. In other words, the model may remove some correct entities incorrectly. But it is encouraging to see the novel type filtering model gets a better result than traditional type filtering model (it increases by 26.38% in MLE and 26.27% in χ^2).

6 Conclusions and Future Works

For the issue of related entity finding, entity ranking is still an important issue. While some entities do not confirm to the required entity type and will affect the ranking result, filtering wrong type entities is essential. First, we parse the original narrative and acquire the noun and verb phrases as the new query. Then we make use of a novel type filtering model and the traditional type filtering model to filter entities respectively. In the experiment section, we choose 15 topics which target entity types are Person, Product and Organization as our test topics. We compare the experiment results and find: (i) the approach that we acquire a new query is

feasible (ii) using our novel type filtering model gets a better result than using the traditional type filtering model whatever in precision or recall. We also see the problem of our type filtering model: compare to the pure co-occurrence model (in $R@2000$), it reduces the recall. It indicates some correct entities are removed incorrectly and the model needs to be improved further.

In this paper we only argue the problem of wrong entity type filtering, while the wrong relation also affects the entity ranking result. As future work, we plan to investigate two issues: (i) how can we optimize our type filtering model to improve recall and precision further (ii) how can we use relation filtering to further optimize the result of entity ranking.

Acknowledgement

This work is supported by the Fundamental Research Funds for the Central Universities of China (Program No.2011JBM231).

Bibliography

- [1] K. Balog, A. P. de Vries, P. Serdyukov, P. Thomas, and T. Westerveld, Overview of the TREC 2009 entity track, *TREC 09*, 2009.
- [2] E. Riloff, Automatically generating extraction patterns from untagged text, *AAAI*, 2:1044-1049, 1996.
- [3] E. M. Voorhees, Overview of the TREC 2002 Question Answering Track, *TREC 02*, 115-123, 2009.
- [4] K. Balog, People Search in the Enterprise. PhD thesis, University of Amsterdam, 2008.
- [5] K. Balog, L. Azzopardi, and M. de Rijke, A language modeling framework for expert finding, *Inf. Proc. and Man.*, 45(1): 1-19, 2009.
- [6] Jovan Pehcevski, James A. Thom, Anne-Marie Vercoestre, Vladimir Naumovski, Entity ranking in Wikipedia: utilising categories, links and topic difficulty prediction, *Information Retrieval*, 13(5):568-600, 2010.
- [7] W. Zheng, S. Gottipati, J. Jiang, and H. Fang, UDEL/SMU at TREC 2009 Entity Track, *TREC 09*, 2009.
- [8] Q. Yang, P. Jiang, C. Zhang, and Z. Niu, Experiments on related entity finding track at TREC 2009, *TREC 09*, 2009.
- [9] Y. Wu and H. Kashioka, NiCT at TREC 2009: Employing three models for Entity Ranking Track, *TREC 09*, 2009.
- [10] Y. Fang et al, Entity retrieval with hierarchical relevance model, *TREC 09*, 2009.
- [11] R. Kaptein, P. Serdyukov, A. de Vries, and J. Kamps, Entity ranking using Wikipedia as a pivot, *CIKM*, 2010.
- [12] <http://lemurproject.org>.
- [13] M. Bron and K. Balog and M. de Rijke, Ranking related entities: components and analyses, *CIKM*, 2010.

- [14] P. Serdyukov and A. de Vries, Delft university at the TREC 2009 Entity Track: Ranking wikipedia entities, *TREC 09*, 2009.
- [15] F. Song and W. B. Croft, A general language model for information retrieval, *CIKM 99*, 77-82, 1999.
- [16] C. D. Manning and H. Schuetze, Foundations of Statistical Natural Language Processing, *The MIT Press*, 1999.
- [17] A. de Vries et al, Overview of the INEX 2007 Entity Ranking Track, 245-251, 2007.
- [18] E. Brill, Transformation-based error-driven learning and natural language processing: a case study in part of speech tagging, *Computational Linguistics*, 21(4): 543-565, 1995.
- [19] L. Ramshaw, and M. Marcus, Text Chunking Using Transformation-Based Learning, *Proc. of the Third ACL Workshop on Very Large Corpora*, MIT, 1995.