ABSTRACT
This document describes the work performed by the TALP Research Center, UPC in its first participation at ERD 2014 short text evaluation track. The objective of this evaluation track is to recognize mentions of entities in a given short text, disambiguate them and map them to the entities in a given collection of knowledge base. To this end, we presented our system taking advantage of a topic modeling approach to rank candidates of each entity mentions occurring in the query text.

Categories and Subject Descriptors
I.2.7 [Artificial Intelligence]: Natural Language Processing; H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval

General Terms
Algorithms, Experimentation, Performance

Keywords
Named Entity, Named Entity Disambiguation, linked data, Topic Modeling

1. INTRODUCTION

The nature of human language contains ambiguity, means a word (or a sequence of words) can interpret in different ways depending on the context in which it appears. This ambiguity can make difficult for natural language processors to analyze unstructured texts. Word Sense Disambiguation (WSD) aims to overcome this challenge through discovering the most appropriate sense of a word using different classification approaches. For instance, the word bar in the sentence “the piece of bar” might include two distinct senses, one as a piece of chocolate and next as a metal bar. WSD is an NP-complete problem [10] which is, by comparing, to NP-completeness in complexity theory, a problem whose difficulty is equivalent to solving central problems of Artificial Intelligence (AI), e.g. the Turing Test [20]. The Named Entity Disambiguation (NED) is a parallel task to WSD whose aim is to discover true reference of a named entity mention occurring in unstructured text. NED is usually associated with Name Entity Recognition (NER), a subtask of information extraction aiming to assign predefined categories such as person, location, organization, etc. to the atomic entities in the text. This paper focuses to solve the ambiguity of Named Entities (NE) using a topic modeling approach.

1.1 Motivation
NED is substantial task to create and augment the Knowledge Bases (KB). A KB is appropriate for both human and machine readability, involved to keep and categorize entities and their relations. The ability to create a KB would greatly enhance the human reader’s capability to obtain large amounts of important knowledge about a subject in a significantly shorter span of time than reading all of the source material. The manual creation of KB is an expensive and time consuming effort [16], which must be repeated every time the disambiguation scenario changes (e.g., in the presence of new domains, different languages, and even sense inventories). The high cost of manual elicitation to create KB forces toward automatic acquisition from text. The NED task is also useful for natural language understanding [7], e.g. it is used to enrich texts with semantic information to provide useful meta-information to unstructured texts. One usage is the automatic link generation to refer entities in news articles, e.g. [13] proposed to parse new WP articles and to automatically create tags and links to other articles about relevant entities. Furthermore, NED can be used in e-mail clients to process messages and recognize references to people in the contact list, current tasks or upcoming events in the calendar. Disambiguating entities is also necessary for most knowledge discovery tasks focusing on real-life entities, such as firms, e.g. to monitor events like firms incorporation or new product releases as done by [19]. A system must be able to accurately recognize references to firms, even if they are being referred to by highly ambiguous acronyms like ABC with roughly 100 different word senses in the English WP.

1.2 Problem Definition and Challenges

References to entities such as people, places and organizations are difficult to track in text, because entities can be referred to by many mention strings (synonymy), and the same mention string may be used to refer to multiple entities (ambiguity). For instance, “George Yardley” might refer to either the Scottish former footballer, or the American basketball player (ambiguity), who is also known by nicknames such as “Yardbird” or shortly “Bird” (synonymy). The ambiguity can be more challenging, e.g. in the sentence “they have big country in this nba match.”, the surface form
“big country” is referring to “Bryant Reeves”, the NBA professional basketball player. In Discussion Fora (DF) such as blogs, etc. the texts might contain grammatical irregularities which make the NED even harder, e.g. consider the sentence “James Hatfield is working with Kirk Hammett”. The surface form “James Hatfield” can be referred to American author but the correct grammatical form of “Hatfield” is “Hetfield” referring to main songwriter and co-founder of heavy metal band Metallica. These synonymy and ambiguity problems make difficult for natural language processors to collect and exploit information about entities across documents without first linking the mentions to a KB. A Disambiguator must use available information, such as the context of the entity mention or information from the KB, to decide which entity is being referred to in the text. This task becomes increasingly difficult when an entity is not referred to by its full name, but by partial names, acronyms or other name variations (e.g., D. Cameron, David W. D. Cameron, Cameron, D. C., etc.). The task heavily relies on knowledge. Without knowledge, it would be impossible for both humans and machines to identify the meaning.

This paper presents our participation in the first Entity Recognition and Disambiguation (ERD) short text evaluation track [3]. We apply a topic modeling approach in order to disambiguate the entities.

2. LITERATURE REVIEW

Recently, much research is performed in the NED scope. Due to the diversity of applications, the actual task and available background information for entity disambiguation often varies. However, there are two common basic approaches to this problem. The first approach uses similarity measures to texts with unstructured knowledge about entities, such as WP articles. The second approach utilizes semi-structured data such as Freebase for disambiguation.

2.1 Development of NED

There are a number of studies on named entity disambiguation. [1] used a Bag of Words (BOW) model to resolve ambiguities among names of people. [11] improved the performance of personal names disambiguation by adding biographic features. [5] trained a Maximum Entropy model with Web Features, Overlap Features, and some other features to judge whether two names refer to the same individual. [17] developed features to represent the context of an ambiguous name with the statistically significant bi-grams. These methods determined to which entity a specific name refer by measuring the similarity between the context of the specific name and the context of the entities. They measured similarity with a BOW model. [9] proposed an alternative similarity metric based on the probability of walking from one ambiguous name to another in a random walk within the social network constructed from all documents. [15] considered extended similarity metrics for documents and other objects embedded in graphs, facilitated via a lazy graph walk, and used it to disambiguate names in email documents. [2] disambiguated web appearances of people based on the link structure of Web pages. These methods tried to add background knowledge via social networks. Social networks can capture the relatedness between terms. [6] proposed to use WP as the background knowledge for disambiguation. By leveraging WP’s semantic knowledge like social relatedness between named entities and associative relatedness between concepts, they can measure the similarity between entities more accurately. [4] and [18] used WP’s category information in the disambiguation process. Using different background knowledge, researcher may find different efficient features for disambiguation.

Hence researchers have proposed so many efficient features for disambiguation, it is important to combine these features to improve the system performance. Some researchers combine features by manual rules or weights. However, it is not always convenient to directly use these rules or weights in another data set. Some researchers also try to use machine learning methods to combine the features. [14] used typical classifiers such as Naive Bayes, C4.5 and SVM to combine features. They trained a two-class classifier to judge whether a candidate is a correct target. And then when they try to do disambiguation for one query, each candidate will be classified into the two classes: correct target or incorrect target. Finally the candidate answer with the highest confidence is selected as the target if there are more than one candidates classified as answers.

2.2 ERD Evaluation Framework

A recent notable contribution to research in the field of NED was made by the participants of the Entity Recognition and Disambiguation (ERD). As the most structured challenging competition, ERD has commenced its activity since 2014 in the content of SIGIR conference where the organizers intended to improve the results of search engines based on the recognized entities in the searched queries. The objective of an Entity Recognition and Disambiguation (ERD) system is to recognize mentions of entities in a given text, disambiguate them, and map them to the entities in a given entity collection or knowledge base. The Challenge is composed of two parallel tracks. In the “long text” track, the challenge targets are pages crawled from the Web; these contain documents that are meant to be easily understandable by humans. The “short text” track, on the other hand, consists of web search queries that are intended for a machine. As a result, the text is typically short and often lacks proper punctuation and capitalization.

3. METHODOLOGY

The method proposed in this paper follows the typical architecture in the state of the art (Figure 1). Briefly, given a query consisting of just a short fragment of text (usually, includes 3 ~ 10 words), the system starts by process the query text (Query Preprocessing step). Subsequently, those KB nodes which can be potential candidates to be the correct entities are selected (candidate generation step). Finally, the candidates are ranked in a top-down hierar-

1http://web-ngram.research.microsoft.com/ERD2014/
2The ERD organized and sponsored by Google and Microsoft.
3http://sigir.org/sigir2014/
chy and the candidates having the highest order are selected as the correct references of the entity mentions occurring in the query text (candidate ranking step). The final task (candidate ranking) is the most challenging and highly crucial among steps above. In order to rank candidates, we apply a topic modeling approach.

Details of each step are provided next.

### 3.1 Query Pre-processing

Usually, named entities appear in the text in different forms. A Surface Form (entity mention) of a named entities is the form of that entity as it appears in a text. The objective of this step is to reveal the most informative and discriminative surface forms occurring in the query text. This step also includes the process of the query for possible expansion of the query text in the case of abbreviation occurrences (e.g. US states).

**Gazetteer-based Expansion.**

Sometimes, query text contains abbreviations. In these occasions, auxiliary gazetteers are beneficial to map the pairs of \(\text{abbreviation, expansion}\) such as the US states, (e.g. the pair \(\langle\text{CA},\text{California}\rangle\) or \(\langle\text{MD, Maryland}\rangle\)), and country abbreviations, (e.g. the pairs \(\langle\text{UK, United Kingdom}\rangle,\langle\text{US, United States}\rangle\) or \(\langle\text{UAE, United Arab Emirates}\rangle\)). We applied this technique to obtain more informative and discriminative query text (Table 1.i).

**Surface Form Extraction.**

In this subsection we present how the text parsing and surface form selection process is carried out. To extract the surface forms in a text, a “sliding window” approach is applied. The text is parsed from beginning to end, and in each step the sliding window selects a small number of consecutive words. The sliding window size is set to a maximum size (the maximum number of words to be selected) at the beginning. In each step the selected word set from the window is searched in the KB. If a surface form is found then it is added to the list of discovered surface forms, otherwise the window size is decreased by one from the right end of the window and the new surface form candidate is searched again in the KB until a match is found. In the case a match is found, the sliding window size set to the maximum size again and slide over the matching words to the next word in line. If no match is found when the window size is shrunk to one word, then the window is again set to the maximum size and slide over to the next word in line. In this approach, the maximum size of the sliding window must be set to a realistic one. The size should neither be too small so that the entities or surface forms that have long labels are not missed, nor too large so that the processing time is not too high. As the query text in the experiment is short (usually, includes 3 ~ 10 words), we decided to use a sliding window with the maximum size of whole text sequence (Table 1.ii and Table 1.iii).

### 3.2 Candidate Generation

The system is required to find the correct reference entry of each surface forms of the query text from the KB. As KB usually contains a large number of entries, it is desirable to avoid brute force comparisons between a particular query and all KB entries and to reduce the search space of potential candidates. To this end, the system starts to generate all possible candidates that can be a potential correct reference of each surface form. Our priority, however, is to generate a large candidate set instead of a smaller one in order to increase recall [12, 8].

Given a particular query, a set of candidates is found for each surface form occurring in the query text by retrieving those entries from the KB whose names are similar enough, using Dice coefficient, to each surface form. In our experiments we used a similarity threshold of 0.6 to extract the candidates from KB (Table 1.iv).

### 3.3 Candidate Ranking

In EL, query expansion techniques are alike across systems, and KB node candidate generation methods normally achieve more than 95% recall. Therefore, the most crucial step is ranking the KB candidates and selecting the best node. In this step, we applied a topic modeling approach to rank the candidates.

**Topic Modeling-based Ranking.**

This module sorts the retrieved candidates according to the likelihood of being the correct referent. We employs VSM, in which a vectorial representation of the query text is compared with the vectorial representation of the reference KB candidates. The vector space domain consist of the whole set of words within the query text and the rank consists of their tf/idf computed against the set of candidate WP documents. We use cosine similarity. In addition, in order to reduce dimensionality we apply LSI. The module iterates the procedure for each surface form of the query text to match the best candidate for each of them (Table 1.v).

### 4. EVALUATION FRAMEWORK

We participated in the framework of the ERD 2014 short text evaluation track\(^4\).

---

\(^4\)http://web-ngram.research.microsoft.com/
In the short text track, given a short fragment of text for each query, the ERD system should use all available context to choose the correct reference of each entity surface form occurring in the query text. The reference KB used in this track includes hundreds of thousands of entities based on articles from an September 2013 dump of Freebase\textsuperscript{5}. In this track, the entities in the reference KB are the only entities that are evaluated (filtering out those entries having general definition e.g. the KB entry “Apple” as a fruit), keeping only those that have English Wikipedia pages associated with them. In addition, the type information in Freebase is used to further filter unwanted entities. The public evaluation set is composed of a subset of web search queries from past TREC competitions. Each query will be mapped to a set of valid interpretations, based on majority agreement among human judges. Another set of web search queries, annotated using the similar guidelines, is used for the final evaluation. In training phase, a set of around 100 queries together with their corresponding annotations was provided. In each of development and testing phases, 500 queries (total 1000 queries) were assessed.

5. RESULTS AND ANALYSIS

We participated at the Entity Recognition and Disambiguation (ERD 2014) short text evaluation track. In the short text track, given a query text for each query, the ERD system should use all available context to select the correct KB candidate of each entity surface form occurring in the query text.

Nineteen teams participated in the ERD 2014 short text evaluation track. Each team were allowed to assess several systems during training and development phases but only their primary system was tested during the evaluation phase. The ERD systems were evaluated through online public-web query requests. In the ERD short text evaluation track, each query had to be responded in a time span less than twenty seconds, otherwise, the corresponding query is rejected. The best team in this evaluation achieved 0.685 F1 of the final evaluation. Although we faced to many technical problems (almost server and connection problems) during the training, development, and evaluation phases, and these problems strongly reduced the performance and final score of our system in the evaluation phase, however the system could achieve 0.450 F1 of the evaluation phase. Due to this technical problem and their effects on the system, our proposed approach could not be exactly assessed. In the evaluation phase, our system dropped 6 timeout queries out of whole 500 evaluation queries in total.

Apart from the problems that came up to us, this baseline system provided us a substantial framework in order to struggle with the task challenges and participating in next ERD evaluations.

6. CONCLUSIONS AND FUTURE WORK

In our participation in the first Entity Recognition and Disambiguation (ERD 2014), we applied a topic modeling approach in order to rank the candidates of each surface form occurring in a query short fragment of text. Although, because of many technical issues, we could not assess all capabilities and performance of our system, but it was an appropriate baseline to improve the system in the future.

As a future work, we hang on to improve the system results through profound analyzing of semantics of the query text in order to increase the accuracy of the task. The deep semantic analysis of the text is beneficial when the queries are highly ambiguous to realize their correct references.

7. ACKNOWLEDGMENTS

This work has been produced with the support of the SKATER project (TIN2012-38584-C06-01).

8. REFERENCES

[18] M. P. R. Buneanu. Using encyclopedic knowledge for named entity disambiguation. In Proceedings of the the 11th Conference of the European Chapter of the Association for...