Chinese Term Extraction Using Different Types of Relevance

Yuhang Yang[1], Tiejun Zhao[1], Qin Lu[2], Dequan Zheng[1] and Hao Yu[1]
[1]School of Computer Science and Technology,
Harbin Institute of Technology, Harbin 150001, China
{yhyang,tjzhao,dqzheng,yu}@mtlab.hit.edu.cn
[2]Department of Computing,
The Hong Kong Polytechnic University, Hong Kong, China
csluqin@comp.polyu.edu.hk

Abstract

This paper presents a new term extraction approach using relevance between term candidates calculated by a link analysis based method. Different types of relevance are used separately or jointly for term verification. The proposed approach requires no prior domain knowledge and no adaptation for new domains. Consequently, the method can be used in any domain corpus and it is especially useful for resource-limited domains. Evaluations conducted on two different domains for Chinese term extraction show significant improvements over existing techniques and also verify the efficiency and relative domain independent nature of the approach.

1 Introduction

Terms are the lexical units to represent the most fundamental knowledge of a domain. Term extraction is an essential task in domain knowledge acquisition which can be used for lexicon update, domain ontology construction, etc. Term extraction involves two steps. The first step extracts candidates by unithood calculation to qualify a string as a valid term. The second step verifies them through termhood measures (Kageura and Umino, 1996) to validate their domain specificity.

Many previous studies are conducted on term candidate extraction. Other tasks such as named entity recognition, meaningful word extraction and unknown word detection, use techniques similar to that for term candidate extraction. But, their focuses are not on domain specificity. This study focuses on the verification of candidates by termhood calculation.

Relevance between term candidates and documents is the most popular feature used for term verification such as *TF-IDF* (Salton and McGill, 1983; Frank, 1999) and *Inter-Domain Entropy* (Chang, 2005), which are all based on the hypothesis that "if a candidate occurs frequently in a few documents of a domain, it is likely a term". Limited distribution information of term candidates in different documents often limits the ability of such algorithms to distinguish terms from non-terms. There are also attempts to use prior domain specific knowledge and annotated corpora for term verification. *TV_ConSem* (Ji and Lu, 2007) calculates the percentage of context words in a domain lexicon using both frequency information and semantic information. However, this technique requires a domain lexicon whose size and quality have great impact on the performance of the algorithm. Some supervised learning approaches have been applied to protein/gene name recognition (Zhou et al., 2005) and Chinese new word identification (Li et al., 2004) using *SVM* classifiers (Vapnik, 1995) which also require large domain corpora and annotations. The latest work by Yang (2008) applied the relevance between term candidates and sentences by using the link analysis approach based on the *HITS* algorithm to achieve better performance.

In this work, a new feature on the relevance between different term candidates is integrated with other features to validate their domain specificity. The relevance between candidate terms may be useful to identify domain specific terms based on two assumptions. First, terms are more likely to occur with other terms in order to express domain information. Second, term candidates extracted from domain corpora are likely

Proceedings of the ACL-IJCNLP 2009 Conference Short Papers, pages 213–216,
Suntec, Singapore, 4 August 2009. ©2009 ACL and AFNLP

to be domain specific. Previous work by (e.g. Ji and Lu, 2007) uses similar information by comparing the context to an existing large domain lexicon. In this study, the relevance between term candidates are iteratively calculated by graphs using link analysis algorithm to avoid the dependency on prior domain knowledge.

The rest of the paper is organized as follows. Section 2 describes the proposed algorithms. Section 3 explains the experiments and the performance evaluation. Section 4 concludes and presents the future plans.

2 Methodology

This study assumes the availability of term candidates since the focus is on term verification by termhood calculation. Three types of relevance are first calculated including (1) the term candidate relevance, *CC*; (2) the candidate to sentence relevance, *CS*; and the candidates to document relevance, *CD*. Terms are then verified by using different types of relevance.

2.1 Relevance between Term Candidates

Based on the assumptions that term candidates are likely to be used together in order to represent a particular domain concept, relevance of term candidates can be represented by graphs in a domain corpus. In this study, *CC* is defined as their co-occurrence in the same sentence of the domain corpus. For each document, a graph of term candidates is first constructed. In the graph, a node is a term candidate. If two term candidates TC_1 and TC_2 occur in the same sentence, two directional links between TC_1 to TC_2 are given to indicate their mutually related. Candidates with overlapped substrings are not removed which means long terms can be linked to their components if the components are also candidates.

After graph construction, the term candidate relevance, *CC*, is then iteratively calculated using the PageRank algorithm (Page et al. 1998) originally proposed for information retrieval. PageRank assumes that the more a node is connected to other nodes, it is more likely to be a salient node. The algorithm assigns the significance score to each node according to the number of nodes linking to it as well as the significance of the nodes. The PageRank calculation *PR* of a node *A* is shown as follows:

$$PR(A) = (1-d) + d\left(\frac{PR(B_1)}{C(B_1)} + \frac{PR(B_2)}{C(B_2)} + \ldots + \frac{PR(B_t)}{C(B_t)}\right) \qquad (1)$$

where B_1, B_2,…, B_t are all nodes linked to node *A*; $C(B_i)$ is the number of outgoing links from node B_i; *d* is the factor to avoid loop trap in the graphic structure. *d* is set to 0.85 as suggested in (Page et al., 1998). Initially, all *PR* weights are set to 1. The weight score of each node are obtained by (1), iteratively. The significance of each term candidate in the domain specific corpus is then derived based on the significance of other candidates it co-occurred with. The *CC* weight of term candidate TC_i is given by its *PR* value after *k* iterations, a parameter to be determined experimentally.

2.2 Relevance between Term Candidates and Sentences

A domain specific term is more likely to be contained in domain relevant sentences. Relevance between term candidate and sentences, referred to as *CS*, is calculated using the *TV_HITS* (Term Verification – HITS) algorithm proposed in (Yang et al., 2008) based on Hyperlink-Induced Topic Search (*HITS*) algorithm (Kleinberg, 1997). In *TV_HITS*, a good hub in the domain corpus is a sentence that contains many good authorities; a good authority is a term candidate that is contained in many good hubs.

In *TV_HITS*, a node *p* can either be a sentence or a term candidate. If a term candidate *TC* is contained in a sentence *Sen* of the domain corpus, there is a directional link from *Sen* to *TC*. *TV_HITS* then makes use of the relationship between candidates and sentences via an iterative process to update *CS* weight for each *TC*.

Let $V^A(w(p_1)^A, w(p_2)^A, \ldots, w(p_n)^A)$ denote the authority vector and $V^H(w(p_1)^H, w(p_2)^H, \ldots, w(p_n)^H)$ denote the hub vector. V^A and V^H are initialized to (1, 1,…, 1). Given weights V^A and V^H with a directional link $p \rightarrow q$, $w(q)^A$ and $w(p)^H$ are updated by using the *I* operation(an in-pointer to a node) and the *O* operation(an out-pointer to a node) shown as follows. The *CS* weight of term candidate TC_i is given by its $w(q)^A$ value after iteration.

$$I \text{ operation: } w(q)^A = \sum_{p \rightarrow q \in E} w(p)^H \qquad (2)$$

$$O \text{ operation: } w(p)^H = \sum_{p \rightarrow q \in E} w(q)^A \qquad (3)$$

2.3 Relevance between Term Candidates and Documents

The relevance between term candidates and documents is used in many term extraction algorithms. The relevance is measured by the *TF-IDF* value according to the following equations:

$$TFIDF(TC_i) = TF(TC_i) \cdot IDF(TC_i) \quad (4)$$

$$IDF(TC_i) = \log(\frac{|D|}{DF(TC_i)}) \quad (5)$$

where *TF(TC$_i$)* is the number of times term candidate *TC$_i$* occurs in the domain corpus, *DF(TC$_i$)* is the number of documents in which *TC$_i$* occurs at least once, |*D*| is the total number of documents in the corpus, *IDF(TC$_i$)* is the inverse document frequency which can be calculated from the document frequency.

2.4 Combination of Relevance

To evaluate the effective of the different types of relevance, they are combined in different ways in the evaluation. Term candidates are then ranked according to the corresponding termhood values *Th(TC)* and the top ranked candidates are considered terms.

For each document *D$_j$* in the domain corpus where a term candidate *TC$_i$* occurs, there is *CC$_{ij}$* weight and a *CS$_{ij}$* weight. When features *CC* and *CS* are used separately, termhood *Th$_{CC}$(TC$_i$)* and *Th$_{CS}$(TC$_i$)* are calculated by averaging *CC$_{ij}$* and *CS$_{ij}$*, respectively. Termhood of different combinations are given in formula (6) to (9). *R(TC$_i$)* denotes the ranking position of *TC$_i$*.

$$Th_{CC+CS}(TC_i) = \frac{1}{R_{CC}(TC_i)} + \frac{1}{R_{CS}(TC_i)} \quad (6)$$

$$Th_{CC+CD}(TC_i) = (\sum_j CC_{ij}) \log(\frac{|D|}{DF_C}) \quad (7)$$

$$Th_{CS+CD}(TC_i) = (\sum_j CS_{ij}) \log(\frac{|D|}{DF_C}) \quad (8)$$

$$Th_{CC+CS+CD}(TC_i) = \frac{1}{R_{CC+CD}(TC_i)} + \frac{1}{R_{CS+CD}(TC_i)} \quad (9)$$

3 Performance Evaluation

3.1 Data Preparation

To evaluate the performance of the proposed relevance measures for Chinese in different domains, experiments are conducted on two separate domain corpora *Corpus*$_{IT}$ and *Corpus*$_{Legal}$., respectively. Corpus$_{IT}$ includes academic papers of 6.64M in size from Chinese IT journals between 1998 and 2000. *Corpus*$_{Legal}$ includes the complete set of official Chinese constitutional law articles and Economics/Finance law articles of 1.04M in size (http://www.law-lib.com/).

For comparison to previous work, all term candidates are extracted from the same domain corpora using the delimiter based algorithm *TCE_DI* (Term Candidate Extraction – Delimiter Identification) which is efficient according to (Yang et al., 2008). In *TCE_DI*, term delimiters are identified first. Words between delimiters are then taken as term candidates.

The performances are evaluated in terms of precision (P), recall (R) and *F*-value (F). Since the corpora are relatively large, sampling is used for evaluation based on fixed interval of 1 in each 10 ranked results. The verification of all the sampled data is carried out manually by two experts independently. To evaluate the recall, a set of correct terms which are manually verified from the extracted terms by different methods is constructed as the standard answer. The answer set is certainly not complete. But it is useful as a performance indication for comparison since it is fair to all algorithms.

3.2 Evaluation on Term Extraction

For comparison, three reference algorithms are used in the evaluation. The first algorithm is *TV_LinkA* which takes *CS* and *CD* into consideration and performs well (Yang et al., 2008). The second one is a supervised learning approach based on a *SVM* classifier, SVM^{light} (Joachims, 1999). Internal and external features are used by SVM^{light}. The third algorithm is the popular used *TF-IDF* algorithm. All the reference algorithms require no training except SVM^{light}. Two training sets containing thousands of positive and negative examples from IT domain and legal domain are constructed for the *SVM* classifier. The training and testing sets are not overlapped.

Table 1 and Table 2 show the performance of the proposed algorithms using different features for IT domain and legal domain, respectively. The algorithm using *CD* alone is the same as the *TF-IDF* algorithm. The algorithm using *CS* and *CD* is the *TV_LinkA* algorithm.

Algorithms	Precision (%)	Recall (%)	F-value (%)
SVM	63.6	49.5	55.6
CC	47.1	36.5	41.2
CS	65.6	51	57.4
CD(TF-IDF)	64.8	50.4	56.7
CC+CS	80.4	62.5	70.3
CC+CD	49	38.1	42.9
CS+CD (TV_LinkA)	75.4	58.6	66
CC+CS+CD	82.8	64.4	72.4

Table 1. Performance on IT Domain

Algorithms	Precision (%)	Recall (%)	F-value (%)
SVM	60.1	54.2	57.3
CC	45.2	40.3	42.6
CS	70.5	40.1	51.1
CD(TF-IDF)	59.4	52.9	56
CC+CS	64.2	49.9	56.1
CC+CD	48.4	43.1	45.6
CS+CD (TV_LinkA)	67.4	60.1	63.5
CC+CS+CD	70.2	62.6	66.2

Table 2. Performance on Legal Domain

Table 1 and Table 2 show that the proposed algorithms achieve similar performance on both domains. The proposed algorithm using all three features (*CC+CS+CD*) performs the best. The results confirm that the proposed approach are quite stable across domains and the relevance between candidates are efficient for improving performance of term extraction in different domains. The algorithm using *CC* only does not achieve good performance. Neither does *CC+CS*. The main reason is that the term candidates used in the experiments are extracted using the *TCE_DI* algorithm which can extract candidates with low statistical significance. *TCE_DI* provides a better compromise between recall and precision. *CC* alone is vulnerable to noisy candidates since it relies on the relevance between candidates themselves. However, as an additional feature to the combined use of *CS* and *CD* (*TV_LinkA*), improvement of over 10% on F-value is obtained for the IT domain, and 5% for the legal domain. This is because the noise data are eliminated by *CS* and *CD*, and *CC* help to identify additional terms that may not be statistically significant.

4 Conclusion and Future Work

In conclusion, this paper exploits the relevance between term candidates as an additional feature for term extraction approach. The proposed approach requires no prior domain knowledge and no adaptation for new domains. Experiments for term extraction are conducted on IT domain and legal domain, respectively. Evaluations indicate that the proposed algorithm using different types of relevance achieves the best performance in both domains without training.

In this work, only co-occurrence in a sentence is used as the relevance between term candidates. Other features such as syntactic relations can also be exploited. The performance may be further improved by using more efficient combination strategies. It would also be interesting to apply this approach to other languages such as English.

Acknowledgement: The project is partially supported by the Hong Kong Polytechnic University (PolyU CRG G-U297)

References

Chang Jing-Shin. 2005. Domain Specific Word Extraction from Hierarchical Web Documents: A First Step toward Building Lexicon Trees from Web Corpora. In *Proc of the 4th SIGHAN Workshop on Chinese Language Learning*: 64-71.

Eibe Frank, Gordon. W. Paynter, Ian H. Witten, Carl Gutwin, and Craig G. Nevill-Manning. 1999. Domain-specific Keyphrase Extraction. In *Proc.of 16th Int. Joint Conf. on AI, IJCAI-99*: 668-673.

Joachims T. 2000. Estimating the Generalization Performance of a SVM Efficiently. In *Proc. of the Int Conf. on Machine Learning*, Morgan Kaufman, 2000.

Kageura K., and B. Umino. 1996. Methods of automatic term recognition: a review. *Term* 3(2):259-289.

Kleinberg J. 1997. Authoritative sources in a hyperlinked environment. In *Proc. of the 9th ACM-SIAM Symposium on Discrete Algorithms*: 668-677. New Orleans, America, January 1997.

Ji Luning, and Qin Lu. 2007. Chinese Term Extraction Using Window-Based Contextual Information. In *Proc. of CICLing 2007, LNCS 4394*: 62 – 74.

Li Hongqiao, Chang-Ning Huang, Jianfeng Gao, and Xiaozhong Fan. The Use of SVM for Chinese New Word Identification. In *Proc. of the 1st Int.Joint Conf. on NLP (IJCNLP2004)*: 723-732. Hainan Island, China, March 2004.

Salton, G., and McGill, M.J. (1983). *Introduction to Modern Information Retrieval*. McGraw-Hill.

S. Brin, L. Page. The anatomy of a large-scale hypertextual web search engine. The 7th Int. World Wide Web Conf, Brisbane, Australia, April 1998, 107-117.

Vladimir N. Vapnik. 1995. *The Nature of Statistical Learning Theory*. Springer, 1995.

Yang Yuhang, Qin Lu, Tiejun Zhao. (2008). Chinese Term Extraction Using Minimal Resources. The 22nd Int. Conf. on Computational Linguistics (Coling 2008). Manchester, Aug., 2008, 1033-1040.

Zhou GD, Shen D, Zhang J, Su J, and Tan SH. 2005. Recognition of Protein/Gene Names from Text using an Ensemble of Classifiers. *BMC Bioinformatics* 2005, 6(Suppl 1)**:**S7.