View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Waseda University Repository

Unsupervised Word Sense Disambiguation Using Neighborhood Knowledge[1]

Huang Heyan[1,2], Yang Zhizhuo[1,2], and Jian Ping[1,2]

[1]Beijing Engineering Applications Research Center of High Volume Language Information Processing
and Cloud Computing, Beijing Institute of Technology,
No.5 Yard, Zhong Guan Cun South Street Haidian District,Beijing, 100081,China
[2]Department of Computer Science, Beijing Institute of Technology,
No.5 Yard, Zhong Guan Cun South Street Haidian District,Beijing, 100081,China
{hhy63,10907029,pjian}@bit.edu.cn

Abstract. Usually ambiguous words contained in article appear several times. Almost all existing methods for unsupervised word sense disambiguation make use of information contained only in ambiguous sentence. This paper presents a novel approach by considering neighborhood knowledge. The approach can naturally make full use of the within-sentence relationship from the ambiguous sentence and cross-sentence relationship from the neighborhood knowledge. Experimental results indicate the proposed method can significantly outperform the baseline method.

Keywords: Unsupervised WSD, Neighborhood Knowledge, Similarity Measure, Graph-based Ranking Algorithm.

1. Introduction

Word Sense Disambiguation (WSD), the task of indentifying the intended meaning (sense) of words in context is one of the most important problem in natural language processing. Though it is often characterized as an intermediate task rather than an end in itself, it has the potential to improve the performance of many applications including information retrieval, machine translation and so on.

Existing methods conduct WSD usually using only the information contained in the ambiguous sentence to be disambiguated. They utilize context words within predefined window in sentence together with other syntactic information. It has been proved that expanding context window size around the target ambiguous word can help to enhance the WSD performance.

[1] This work is supported by the National Natural Science Foundation of China under Grant No.61132009, the Cloud Computing for High Volume Language Information Intelligent Processing, the Primary Funds of National Defense ,Open Project Program of National Laboratory of Pattern Recognition ,BIT Fundamental Research Projects No.3070012210917

Copyright 2011 by Huang Heyan, Yang Zhizhuo, Jian Ping

However, expanding window size unboundedly will bring not only useful information but also some noise which may finally deteriorate the WSD performance. Can we find other way to expand context words without bringing too much noise?

In this study, we proposed to conduct WSD using collaborative techniques. One common assumption of existing methods is that the sentences are independent of each other, and WSD is carried out separately without interactions among the sentences. However, some ambiguous words contained in article appear more than one time, the multiple sentences contain the same ambiguous word within the article actually have mutual influences and contain useful clues which will helpful to deduce word sense from each other. For example, sentence containing *program design* commonly shares similar topic with the sentence containing *computer program*, thus the meaning of the ambiguous words *program* in those two sentences can be deduced from each other. The idea is borrowed from the observation that ambiguous words appear in topic related sentences contained in the same article often share common meanings. Therefore, we can retrieve a small number of sentences containing the ambiguous word from the same article. These neighbor sentences can be used in the disambiguation process and help to disambiguate word sense for the specified sentence.

This study proposes to construct an appropriate knowledge context for unsupervised WSD method by making use of a few neighbor sentences closed to the ambiguous sentence in the article. The framework for WSD consists of the step of neighborhood knowledge building and the step of word sense disambiguation. In particular, the neighborhood knowledge context is obtained by applying the similarity algorithm on the article. The graph-ranking based algorithm is employed to disambiguate word sense in a specified knowledge context. Instead of leveraging only the word relationships in a single sentence, the algorithm can incorporate the relationship in multiple sentences, thus making use of global information existing in the whole article.

Experiments are carried out on dataset and the results confirm the effectiveness of our method. The neighborhood knowledge can significantly improve the performance of single sentence WSD. Furthermore, how the size of the neighborhood influences the WSD performance is also investigated. It has been reported that a small number of neighbor documents are sufficient to elevate the performance.

The rest of paper is organized as follows: Section 2 briefly introduces the related work. The proposed method is described in detail in Section 3, and experimental results are presented in Section 4. Lastly we conclude this paper in Section 5.

2. Related Work

Generally speaking, Word Sense Disambiguation methods are either knowledge-based or corpus-based. In addition, the latter can further be further divided into two kinds: unsupervised ones and supervised ones. In this paper we focus on unsupervised WSD method.

Knowledge-based method disambiguates words by matching context with information from a prescribed knowledge source. These methods include Lesk’s algorithm (Lesk, 1986), Walker’s algorithm (Walker and Amsler, 1986), Yarowsky's algorithm (Yarowsky, 1992) and so on. Unsupervised methods cluster words into some sets which indicate the same meaning, but they cannot give an exact meaning of the target word, these methods can be categorized into methods based on word clustering (Lin, 1998) and co-occurrence graphs (Widdows and Dorow, 2002).

Supervised method learns form annotated sense examples, the learning algorithms including: SVM (Escudero et al, 2000a), naïve Bayesian learning (Escudero et al, 2000b) and maximum entropy (Tratz et al, 2007). Though corpus-based approach usually has better performance, the mount of words it can disambiguate essentially relies on the size of training corpus, while knowledge-based approach has the advantage of providing larger coverage. Knowledge based methods for word sense disambiguation are usually applicable to all words in the text, while corpus-based techniques usually target only few selected word for which large corpora are available.

More recently, graph-based methods for WSD have gained much attention in the NLP community (Veronis, 2004, Sinha and Mihalcea, 2007, Navigli and Lapata, 2007, Mihalcea, 2005, Agirre E, 2006). The HyperLex (Veronis, 2004) algorithm is entirely corpus-based, which uses small-world properties of co-occurrence graphs. TexRank (Mihalcea, 2005) creates a complete weighted graph formed by the synsets of the words in the input context, and the weight of the edge linking two synsets is calculated by executing Lesk's algorithm. These methods have been proposed to rank word sense based on the "vote" or "recommendations" between each other. When a word sense links to another one, it is basically casting a vote for that word sense. The higher the number of vote that is cast for a sense, the higher the importance of the sense is. Moreover, how important of the word sense is casting the vote determines how important the vote itself is.

All the above WSD methods make use of only information within the ambiguous sentence. It is noteworthy that collaborative techniques have been successfully used in tasks of information filtering (Xue et al., 2005), document summarization (Wan et al., 2007), web mining (Wong et al., 2006) and keyword extraction (wan et al., 2008). To the best of our knowledge, collaborative techniques have been never applied in WSD task. In other words, applying neighbor sentences to improve single sentence WSD has not been investigated yet.

3. Proposed Approach

3.1. Disambiguation Framework

The proposed disambiguation method consists of two steps: neighborhood knowledge building and word sense disambiguation. The first step aims to obtain a few neighbor sentences that topically related to the ambiguous sentence. In the step of word sense disambiguation, the knowledge context expanded by these neighbor sentences is utilized to a better disambiguation of the specified word sense in the ambiguous sentence. The second step can be further divided into two steps: affinity graph building and word sense score computation. First, a global affinity graph G is built based on all sentences related to the ambiguous sentence. The graph will not only reflect the within-sentence relationships (local information) but also the cross-sentence relationships (global information) between words. Then, based on the global affinity graph G, the score of each word sense is computed through the graph-ranking based algorithm. The score calculated here quantifies the importance of the word sense.

The proposed approach first expands knowledge context by finding a few sentences closed to ambiguous sentence, which can provide more knowledge and clues for word sense disambiguation, then an affinity graph is constructed to model both the within-sentence relationships and cross-sentence relationships respectively, finally the saliency score of word sense based on the graph is iteratively computed. Each word gets its saliency score after the algorithm converges and the ambiguous word sense with higher saliency score is chosen to represent the final sense of the ambiguous word.

3.2. Neighborhood Knowledge Building

Usually an ambiguous word occurs in article several times. Which sentences should priority considered to build neighborhood knowledge context? Given an ambiguous sentence, neighborhood knowledge building seeks to select a few nearest neighbors for the sentence from the same article. The neighbor sentences from expanded sentences set can be considered as the expanded knowledge context for the ambiguous sentence.

The performance of neighborhood sentences selection relies on the measurement for sentence similarity evaluation. Given the word collection of a sentence, the semantic similarity between two sentences relies on word's similarity. Distance between words can be computed by either knowledge-based or corpus-based approach. Knowledge-based measure tries to quantify the degree to which two words are semantically related using information drawn from semantic networks. WordNet (Fellbaum, 1998) is a lexical database where each unique meaning of a word is represented by a synonym set. Each synset has a gloss which defines the concept it represents. Synsets are connected to each other through explicit semantic relations that are defined in WordNet. Many graph-based WSD methods have been presented to measure semantic relatedness based on WordNet. Corpus-based methods try to identify the degree of similarity between words using information exclusively derived from large corpora. Such measures as mutual information (Turney 2001) has been proposed to evaluate word semantic similarity based on the co-occurrence information on a large corpus. An advantage of using corpus-based methods is that they reflect the characteristics of the corpus and are potentially better suited for capturing word relation across genres and domains, whereas knowledge-based methods neighbors are corpus-invariant. In this study, we choose the mutual information to compute the semantic similarity between words and as follows:

$$mi(w_k, w_m) = \log \frac{M \times p(w_k, w_m)}{p(w_k) \times p(w_m)} \tag{1}$$

which indicates the degree of statistical dependence between w_k and w_m, Here, M is the total number of words in the corpus and $p(w_k)$ and $p(w_m)$ are respectively the probabilities of the occurrences of w_k and w_m respectively, i.e. $count(w_k)/M$ and $count(w_m)/M$, where $count(w_k)$ and $count(w_m)$ are the frequencies of w_k and w_m. $p(w_k, w_m)$ is the probability of the co-occurrence of w_k and w_m within a window with a predefined size k, i.e. $count(w_m, w_k)/M$, where $count(w_m, w_k)$ is the number of the times w_k and w_m co-occur within the window.

The most popular similarity measure in literature is cosine measure which calculates similarity score though common words between two sentences. However, we notice that ambiguous sentence usually have fewer words than document, and a lot of topic related sentences can barely share same word except some empty words. Therefore, if we use cosine measure to compute similarity between sentences, we cannot differentiate similarity between sentences pairs and thus fall back on a reasonable knowledge context for ambiguous sentence. In this study we proposed to use a new method based on corpus to calculate similarity between sentences, which use mutual information score to record the semantic similarity between words. The formula is as follows:

$$sim(s_i, s_j) = \frac{\sum_{w_k \in s_i} \sum_{w_m \in s_j} mi(w_k, w_m) * tf_{w_k} * isf_{w_k} * tf_{w_m} * isf_{w_m}}{\left\| \vec{s_i} \right\| \times \left\| \vec{s_j} \right\|} \tag{2}$$

where s_i and s_j are the corresponding term vectors of sentences s_i and s_j respectively. The weight associated with term t is calculated with $tf_t * isf_t$, where tf_t is the frequency of term t in the sentence and isf_t is the inverse sentence frequency of term t, i.e. $1 + \log(N / n_t)$, where N is the total number of sentences and n_t is the number of sentences containing term t in a background corpus. w_k and w_m are the corresponding words contained in sentences s_i and s_j respectively. The formula indicates the degree of statistical dependence between sentences s_i and s_j, it will get a large value if the feature words contained in two sentences are strongly related with each other, and vice versa.

In this study, we use corpus-based measure to compute similarity value between the ambiguous sentence s_0 and the sentences in the corpus, and sentences with similarity value exceeding the threshold will be chosen as nearest neighbors for s_0. At last, the expanded sentences set $S = \{s_0, s_1, ..s_{k+1}\}$ is composed by the $k+1$ selected sentences. We will investigate how the size of neighbor sentences k influences the WSD performance in the experiments.

3.3. Word Sense Disambiguation

Affinity graph building: It has been proved that the words within a predefined window around ambiguous word are more effective for disambiguating ambiguous word sense. The words within predefined window around the ambiguous word are chosen as context words. Moreover, because not all words in the document are good indicators for word sense, the words chosen as context words are restricted with syntactic filters, i.e., only the words with a certain part of speech are added. In our experiment, only nouns and verbs are chosen as context words.

Given the expanded sentences set S, let $G = (V, E)$ be an undirected graph reflecting the relationships between words in S. V is the set of vertices and each vertex represents a context word. E is the set of edges, which is a subset of $V \times V$. Each edge in E is associated with an affinity weight between words if their affinity weight exceeds 0. The affinity weight is calculated using formula (1). The links between words in graph G can be categorized into two classes: within-sentence link and cross-sentence link. If both the end words v_i and v_j of a link come from the same sentence, the link is regarded as within-sentence link; otherwise, it is a cross-sentence document link. Actually, the within-sentence link reflects the local information in a sentence, while the cross-sentence link implies the global information in expanded sentences set. In the experiment we will investigate how the combination size of the two types of links between words influence the WSD performance. We use an adjacency matrix $\mathbf{M}$ to describe G with each entry corresponding to the weight of a link in the graph. $\mathbf{M} = (M_{i,j})_{n \times n}$ is defined as following:

$$M_{i,j} = \begin{cases} mi(v_i, v_j), if\,(i \neq j) \\ 0, \text{otherwise} \end{cases} \tag{3}$$

then $\mathbf{M}$ is normalized to make the sum of each row equal to 1. Note that we use the same notation to denote a matrix and its normalized matrix.

Word sense score computation: The computation of important score is based on the following three intuitions: 1) the more neighbors a word sense has, the more important it is; 2) the more important a word sense's neighbor are, the more important it is; 3) the more heavily a word sense is linked with other word senses, the more important it is. Based on the above intuitions, the importance score $ipscore(v_i)$ for word sense v_i can be deduced from those of all other word senses linked with it and further formulated in a recursive form like PageRank (Page et al., 1998) as follows:

$$ipscore(v_i) = (1-d)/n + d * \sum ipscore(v_j) * M_{i,j} \tag{4}$$

where n is the number of vertexes contained in the graph, and d is the dumping factor usually set to 0.85.

We can use the Markov chain model and random reader to formulate the iterative process. In the Markov model, G is treated as the Markov chain, each word as a state and each edge as a transition from one state to another. The random reader continually browses and reads words following the transition matrix $(1-d)\mathbf{e}/n + d\widetilde{\mathbf{M}}$, where e is the vector with all elements equaling to 1. Usually the algorithm will converge after less than 50 times jump. A threshold can be set to control the time of iteration.

4. Experiment

4.1. Experimental Setup

In the experiment, Sogou Chinese collocation relation[2]was used to compute mutual information of words. The collocation corpus involves more than 20 million collocation relations and more than 15000 high-frequency words, which was extracted from over 100 million internet pages on web in October 2006.

To our knowledge, there was no gold standard All-words Chinese word sense disambiguation dataset for evaluation. So we manually annotated an article randomly chosen from internet for evaluation. The article was firstly preprocessed by word segmentation, and then three graduate students were employed to manually label the word sense for each ambiguous word. The extended TongYiCiCiLin[3] was used as the sense inventory. The annotation process last for two days and the annotation conflicts between three annotators were solved by discussion. Finally, the datasets consisted of 76 testing instances for 24 ambiguous words, the average number of meanings for ambiguous word is 2.54, and the average number of sentences for each ambiguous word is 3.16.

Macro-average precision (Liu et al., 2007) was used to evaluate word sense disambiguation performance. Here is the formula: $p_{mar=} \sum_{i=1}^{N} p_i \Big/ N$, $p_i = m_i / n_i$, where N is the number of all ambiguous words, m_i is the time that the ambiguous word is labeled with the correct sense, and n_i is the number of all test instances for this ambiguous word.

[2] http://www.sogou.com/labs/dl/r.html

[3] It is located at http://ir.hit.edu.cn/.

4.2. Evaluation Results

The proposed approach (i.e. CollaDisam) is compared with the baseline method (i.e. OrigDisam) which only relies on the information of ambiguous sentence and applies graph-based algorithm to rank ambiguous word senses.

Table 1: WSD performance.

	OrigDisam	CollaDisam
Average precision	0.4268	0.4594
Improving Performance (%)	3.26	0

Table 1 gives the comparison results of baseline methods and the proposed CollaDisam (within-sentence window size=1, cross-sentence window size=1) methods when using 2 neighbor sentences. We can see from the figures that CollaDisam obtains improvement over OrigDisam method, which indicates its effectiveness.

In order to investigate how the size of the neighborhood influences the WSD performance, we conduct experiments with different values of the neighbor number k and different values of window size n.Figure 1 shows the performance curves for the CollaDisam method. In the figure, k ranges from 0 to 5. Note that when $k = 0$, the CollaDisam method degenerates into baseline OrigDisam method. In the experiment, we set within-sentence and cross-sentence window size the same number n except when $k = 0$. For example, the best performance is achieved when set $n = 1$ and $k = 2$ which means that two words around ambiguous word in three sentences (including one original sentence and two neighbor sentences) will be added to the graph and determine the sense of the ambiguous word.

We can see from the figure that almost all performances of CollaDisam can outperform the baseline OrigDisam method (when $k = 0$), no matter how many neighbor sentences are used. We can also observe that the performances of CollaDisam first increase and then tend to be relatively stable, finally decreased with the increase of k. We can notice that very few neighbors will deteriorate the results, because they cannot provide sufficient knowledge. However, it is not necessary to provide too many neighbors due to the computational complexity problem.

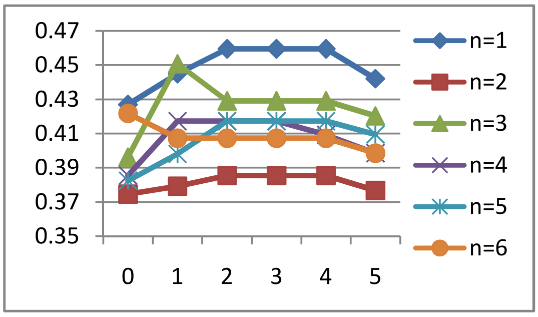

Figure 1: CollaDisam(within-sentence window size= cross-sentence window size) WSD Performance vs. neighbor sentences number

As can be Seen from the figure 1, when k ranges from 2 to 4, the curves tends to be relatively stable, it means that add neighbor sentence into expanded sentences set does not influence the final result, that is partly because some ambiguous words only occur two times in testing article, it seem that all the sentences contain same ambiguous words are useful for determining final sense of ambiguous word and the similarity measure does not play its role, but we argue that similarity measure is extremely important for sentence selection especially for the long article which contain more sentences than short one, since there are some ambiguous word contained in sentence have different meaning compare to other ones, the similarity method can guide us to obtain suitable neighborhood knowledge to improve disambiguation performance.

In order to investigate how the window sizes around the ambiguous word influences WSD performance, we conduct experiment with different values for within-sentence window size $n1$ and cross-sentence window size $n2$. Figure 2 and Figure 3 shows the curves for CollaDisam method with different window sizes. In both figures, window size ranges from 1 to 6 and k represents the different value of neighbor sentences used in experiment. In the experiment, within-sentence window size is fixed at 1 in Figure 2 and cross-sentence window size is fixed at 1 in Figure 3. It can be seen from both figures that almost all curves decline as the window size increasing which means that the context words near ambiguous word have the best disambiguation capacity and enlarging window size will bring noisy information. We can also observe that enlarging neither within-sentence nor cross-sentence window size can improve the WSD performance. The best WSD performance was achieved when both type of window sizes are fixed at 1.

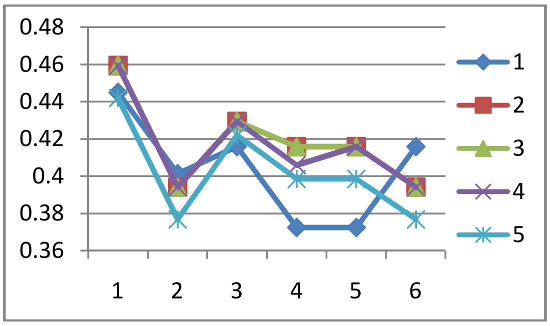

Figure 2: CollaDisam (within-sentence window size=1) WSD Performance vs. cross-sentence window size $n1$

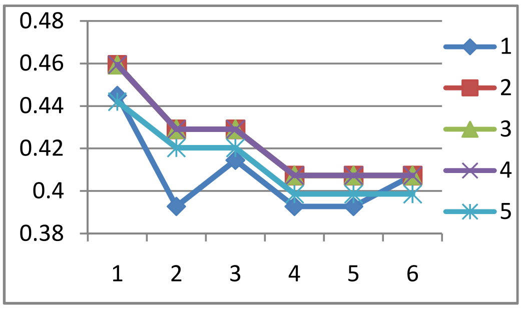

Figure 3: CollaDisam (cross-sentence window size=1) WSD Performance vs. within-sentence window size $n2$

From above experiments, we can draw the conclusion that the two words around ambiguous word have the best disambiguation capability, and the specified words (near ambiguous word) in several similar neighbor sentences is better than the other words (far from ambiguous word) in ambiguous sentence when conducting WSD task. Since the specified words (near ambiguous word in neighbor sentences) would have fewer noisy information than the other words (far from ambiguous word in ambiguous sentence), we can exploit context words in neighbor sentences to improve WSD performance.

5. Conclusion and Future work

In this paper, we proposed a novel approach for unsupervised word sense disambiguation by leveraging neighborhood knowledge of the ambiguous sentence. The within-sentence information and cross-sentence information are incorporated into the graph-based ranking algorithm. The experimental results on dataset demonstrate the good effectiveness of our method.

In current study, we consider sentences in the same article as neighborhood knowledge. Actually the granularity is relatively small. In future work, we will investigate larger granularity including topic and domain, that is to say exploit neighbor sentences in same topic or domain to enhance disambiguation performance, it is very useful for some ambiguous words which appear only one time in small granularity, moreover we will do more experiment in larger dataset to test our method.

References

Michael Lesk. 1986. “Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone”, in Proceedings of the 5th annual international conference on Systems documentation, Toronto, Ontario, Canada, 1986.

Walker D. and Amsler R. 1986. "The Use of Machine Readable Dictionaries in Sublanguage Analysis", in Analyzing Language in Restricted Domains, Grishman and Kittredge (eds), LEA Press, pp. 69-83, 1986.

Yarowsky, David. 1992. "Word sense disambiguation using statistical models of Roget's categories trained on large corpora", in Proceedings of the 14th International Conference on Computational Linguistics (COLING), Nantes, France, 454-460, 1992.

LIN,D. 1998a. Automatic retrieval and clustering of similar words. In Proceedings of the 17th International Conference on Computational linguistics (COLING, Montreal, P.Q., Canada). 768–774.

WIDDOWS, D. AND DOROW, B. 2002. A graph model for unsupervised lexical acquisition. In Proceedings of the 19th International Conference on Computational Linguistics (COLING, Taipei, Taiwan). 1–7.

ESCUDERO, G., M`ARQUEZ, L., AND RIGAU, G. 2000b. Naive Bayes and exemplar-based approaches to word sense disambiguation revisited. In Proceedings of the 14th European Conference on Artificial Intelligence (ECAI, Berlin, Germany). 421–425.

ESCUDERO, G., M`ARQUEZ, L., AND RIGAU, G. 2000. On the portability and tuning of supervised word sense disambiguation. In Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC, Hong Kong, China). 172–180.

TRATZ, S., SANFILIPPO, A., GREGORY, M., CHAPPELL, A., POSSE, C., AND WHITNEY, P. 2007. PNNL: A supervised maximum entropy approach to word sense disambiguation. In Proceedings of the 4th InternationalWorkshop on Semantic Evaluations (SemEval, Prague, Czech Republic). 264–267.

V´ERONIS, J. 2004. Hyperlex: Lexical cartography for information retrieval. Comput. Speech Lang. 18, 3,223–252.

Page, L., Brin, S., Motwani, R., and wingorad, T., 1998.The pagerank citation ranking: Bringing order to the web Technical report, Stanford Digital Library Technologies Project.

R. Mihalcea. 2005. Unsupervised large-vocabulary word sense disambiguation with graph-based algorithms for sequence data labeling. In Proceedings of HLT05, Morristown, NJ, USA.

R. Navigli and M. Lapata. 2007. Graph connectivity measures for unsupervised word sense disambiguation.In IJCAI.

R. Sinha and R. Mihalcea. 2007. Unsupervised graph based word sense disambiguation using measures of word semantic similarity. In Proceedings of ICSC 2007, Irvine, CA, USA.

Agirre E., Lopez de Lacalle O., Martinez D., Soroa A. 2006 Two graph-based algorithms for state-of-the-art WSD Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP). Pages 585–593.

Xiaojun Wan, Jianguo Xiao: CollabRank: Towards a Collaborative Approach to Single-Document Keyphrase Extraction. COLING 2008: 969-976

Xiaojun Wan, Jianwu Yang: CollabSum: exploiting multiple document clustering for collaborative single document summarizations. SIGIR 2007: 143-150

Wong, T.-L.; Lam, W.; and Chan, S.-K. 2006. Collaborative information extraction and mining from multiple web documents. SDM2006.

Xue, G.-R.; Lin, C.; Yang, Q.; Xi, W.; Zeng, H.-J.; Yu, Y.; and Chen, Z.2005. Scalable collaborative filtering using cluster-based smoothing. SIGIR2005.

C. Fellbaum. 1998. WordNet: An Electronic Lexical Database.The MIT Press.

P. Turney. 2001. Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In Proceedings of ECML-2001.

Liu PY，Zhao TJ，Yang MY．mT-WSD：Using search engine for multilingual Chinese-English lexical sample task． SemEval-2007．Madison：Omni Press，2007．169-172．