Wikipedia Mining for An Association Web Thesaurus Construction

Kotaro Nakayama, Takahiro Hara and Shojiro Nishio

Dept. of Multimedia Eng., Graduate School of Information Science and Technology
Osaka University, 1-5 Yamadaoka, Suita, Osaka 565-0871, Japan
TEL: +81-6-6879-4513 FAX: +81-6-6879-4514
{nakayama.kotaro, hara, nishio}@ist.osaka-u.ac.jp

Abstract. Wikipedia has become a huge phenomenon on the WWW. As a corpus for knowledge extraction, it has various impressive characteristics such as a huge amount of articles, live updates, a dense link structure, brief link texts and URL identification for concepts. In this paper, we propose an efficient link mining method pfibf (Path Frequency - Inversed Backward link Frequency) and the extension method "forward / backward link weighting (FB weighting)" in order to construct a huge scale association thesaurus. We proved the effectiveness of our proposed methods compared with other conventional methods such as cooccurrence analysis and TF-IDF.

1 Introduction

A thesaurus is a kind of dictionary that defines semantic relatedness among words. Although the effectiveness is widely proved by various research areas such as natural language processing (NLP) and information retrieval (IR), automated thesaurus dictionary construction (esp. machine-understandable) is one of the most difficult issues. Of course, the simplest way to construct a thesaurus is human-effort. Thousands of contributors have spend much time to construct high quality thesaurus dictionaries in the past. However, since it is difficult to maintain such huge scale thesauri, they do not support new concepts in most cases. Therefore, A large number of studies have been made on automated thesaurus construction based on NLP. However, issues due to complexity of natural language, for instance the ambiguous/synonym term problems still remain on NLP. We still need an effective method to construct a high-quality thesaurus automatically avoiding these problems.

We noticed that Wikipedia, a collaborative wiki-based encyclopedia, is a promising corpus for thesaurus construction. According to statistics of Nature, Wikipedia is about as accurate in covering scientific topics as the Encyclopedia Britannica [1]. It covers concepts of various fields such as Arts, Geography, History, Science, Sports or Games. It contains more than 1.3 million articles (Sept. 2006) and it is becoming larger day by day. Because of the huge scale concept network with a wide-range topic coverage, it is natural to think that Wikipedia can be used as a knowledge extraction corpus. In fact, we already proved that it can be used for accurate association thesaurus construction[2]. Further, several

researches have already proved the importance and effectiveness of Wikipedia Mining[3–6].

However, what seems lacking in these methods is the deep consideration for improving accuracy and scalability. After a number of continuous experiments, we realized that there are possibilities to improve the accuracy because the accuracy changes depending on particular situations. Further, none of previous researches has focused on scalability. WikiRelate [4], for instance, measures the relatedness between two given terms by analyzing (searching) the categories which they belong to. This means that we have to search all combinations of categories of all combinations of terms thus the number of steps for the calculation becomes impossibly huge. To conclude this, we still have to consider the characteristics of Wikipedia and optimize the algorithm in order to extract a huge scale accurate thesaurus.

In this paper, we propose an efficient link structure mining method to construct an association thesaurus from Wikipedia. While almost all researches in this research area analyze the structure of categories in Wikipedia, our proposed method analyzes the link structure among pages because links are explicit relations defined by users.

The rest of this paper is organized as follows. First of all, we introduce a number of researches on automated thesaurus construction in order to make our stance clear. Second, we unveil the characteristics and statistics of Wikipedia in detail. After that, we describe some conventional methods which can be used for Wikipedia mining, and we propose a link mining method "pfibf" and an extension "FB weighting." Then, we describe the results of our experiments. Finally, we draw a conclusion.

2 Wikipedia as a Web Corpus

As we mentioned before, Wikipedia is an attractive Web corpus for knowledge extraction because of the characteristics. In this section, we describe two important characteristics of Wikipedia; a dense link structure and concept identification by URL.

"Dense" means that it has a lot of "inner links," links from pages in Wikipedia to other pages in Wikipedia. Let us show some results of the link structure analysis for Wikipedia (Sept. 2006). Figure 1 shows the distribution of forward and backward links. The statistics unveiled that Wikipedia (Esp. the distribution of backward links) has a typical "power-law" distribution, containing a few nodes with a very high degree and many with a low degree of links. This characteristic, the rich semantic links among concepts, shows us the potential of Wikipedia mining.

Concept identification based on URL is also a key feature on Wikipedia. Ordinary (electric) dictionaries have indexes to find the concepts the user wants to know. However, several concepts are put into one index in most cases. This means that ambiguous terms are listed in one article. This is no problem for humans because it is human readable, but it is not machine understandable. For example, if a sentence "Golden delicious is a kind of apple" exists in an article in a dictionary, humans can immediately understand that "apple" means a fruit. However, it is difficult to analyze for a machine because "apple" is an ambiguous

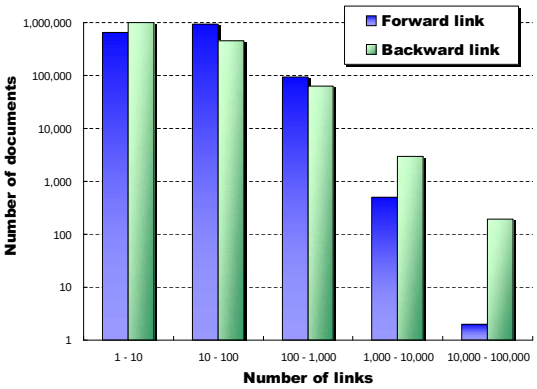

Fig. 1. Distribution of link number.

term and there is no identification information whether it is a fruit or a computer company. On Wikipedia, almost every concept (article/page) has an own URL as an identifier. This means that it is possible to analyze term relations avoiding ambiguous term problems or context problems.

3 Related Works

In this section, we introduce a number of studies for thesaurus construction which relate to our research. After that, we explain how can we apply/extend conventional methods, cooccurrence analysis and TF-IDF (Term Frequency - Inverse Document Frequency) weighting[7] for thesaurus construction.

3.1 Web Structure Mining

One of the most notable differences between an ordinary corpus and a Web corpus is apparently the existence of hyperlinks. Hyperlinks do not just provide a jump function between pages, but have more valuable information. There are two type of links; "forward links" and "backward links." A "forward link" is an outgoing hyperlink from a Web page, an incoming link to a Web page is called "backward link". Recent researches on Web structure mining, such as Google's PageRank[8] and Kleinberg's HITS[9], emphasize the importance of backward links in order to extract objective and trustful data.

By analyzing this information on hyperlinks, we can extract various information such as topic locality[10], site topology, and summary information. Topic locality is the law that web pages which are sharing the same links have more topically similar contents than pages which are not sharing links.

3.2 Wikipedia Mining

"Wikipedia mining" is a new research area which is recently addressed. As we mentioned before, Wikipedia is an invaluable Web corpus for knowledge extrac-

tion. WikiRelate[4] proved that the inversed path length between concepts can be used as a relatedness for two given concepts. However, there are two issues on WikiRelate; the scalability and the accuracy.

The algorithm finds the shortest path between categories which the concepts belong to in a category graph. As a measurement method for two given concepts, it works well. However, it is impossible to extract all related terms for all concepts because we have to search all combinations of category pairs of all concept pairs (1.3 million $\times$ 1.3 million). Furthermore, using the inversed path length as the semantic relatedness is a rough method because categories do not represent the semantic relations in many cases. For instance, the concept "Rook (chess)" is placed in the category "Persian loanwords" with "Pagoda," but the relation is not semantical, it is just a navigational relation.

The accuracy problem of WikiRelate is also mentioned in a Gabrilovich's paper[6]. Gabrilovich proposed a TF-IDF based similarity measurement method for Wikipedia and proved that the accuracy is much better than that of WikiRelate, a category based approach.

3.3 Cooccurrence Analysis

Since the effectiveness of the cooccurrence analysis has been widely proved in the thesaurus construction research area[11], it is possible to apply it to relatedness analysis in Wikipedia. A cooccurrence-based thesaurus represents the similarity between two words as the cosine of the corresponding vectors. Term cooccurrence tc between two terms (t_1 and t_2) can roughly be defined by the following formula:

$$tc(t_1, t_2) = \frac{|D_{t_1} \cap D_{t_2}|}{|D_{t_1} \cup D_{t_2}|}. \quad (1)$$

D_{t_1} is a set of documents which contain term t_1. To measure the similarity of two terms, the number of documents which contain the terms is used. Although the effectiveness has been proved, natural language processing has various accuracy problems due to the difficulty of semantics analysis.

We propose a link cooccurrence analysis for thesaurus construction which uses only the link structure of Web dictionaries in order to avoid the accuracy problems of NLP. Since a Web dictionary is a set of articles (concepts) and links among them, it makes sense to use link cooccurrence as a thesaurus construction method. The formula is basically the same as for the term cooccurrences. The difference is that it uses links in documents instead of terms.

3.4 TF-IDF

TF-IDF[7] is a weighting method that is often used to extract important keywords from a document. TF-IDF uses two measurements; tf (Term Frequency) and idf (Inverse Document Frequency). tf is simply the number of appearances of a term in the document. df describes the number of documents containing the term. In short, the importance of the term basically becomes higher according to the term frequency of the term in the document, and it becomes lower according to the inversed document frequency of the term in the whole collection of documents because idf works as a common terms filter.

Since TF-IDF statistically evaluates the importance of a term to a document in a collection of documents, it also can be used for thesaurus construction because a page corresponds to a concept and the links are semantic associations for other concepts in Web dictionaries. The importance of links to a document can be defined as follows:

$$tfidf(l, d) = tf(l, d) \cdot idf(l), \tag{2}$$

$$idf(l) = \log \frac{N}{df(l)}. \tag{3}$$

$tf()$ denotes the number of appearances of the link l in document d. N is the total number of documents and $df(l)$ returns the number of documents containing the link l. In summary, the importance basically increases according to the link frequency of l in the document d and the inversed document frequency of the link l in the whole collection of documents, thus common links will be filtered by idf. Since a page in Wikipedia corresponds to a concept, by calculating TF-IDF for every link in a page, we can extract a vector of weighted links for the concept. After extracting the vectors for each concept, relatedness between two concepts can be calculated comparing their vectors by using correlation metrics such as cosine metrics.

4 pfibf

$pfibf$ (Path Frequency - Backward link Frequency), the method that we are proposing, is a link structure mining method which is optimized for Wikipedia. While TF-IDF analyzes relationships to neighbor articles (1 hop), $pfibf$ analyzes the relations among nodes in n-hop range. $pfibf$ consists of two factors; pf (Path Frequency) and ibf (Inversed Backward link Frequency). The point is that this is a very balanced method in both scalability and accuracy. In this section, we describe $pfibf$ in detail.

4.1 Basic Strategy

Web-based dictionaries such as Wikipedia consist of a set of articles (concepts) and hyperlinks among them, thus they can be expressed by a graph $G = \{V, E\}$ (V: set of articles, E: set of links). Let us consider how we can measure the relativity between any pair of articles (v_i, v_j). The relativity is assumed to be strongly affected by the following two factors:

- the number of paths from article v_i to v_j,
- the length of each path from article v_i to v_j.

The relativity is strong if there are many paths (sharing of many intermediate articles) between two articles. In addition, the relativity is affected by the path length. In other words, if the articles are placed closely together in the graph G and share hyperlinks to same articles, the relativity is estimated to be higher than farther ones.

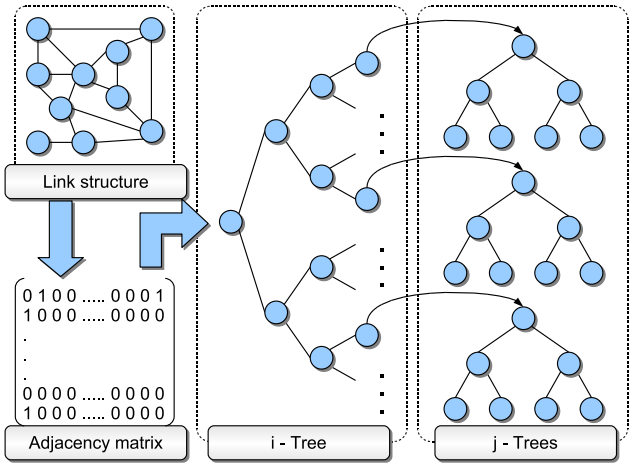

Fig. 2. Dual binary tree for adjacency matrix.

In addition, the number of backward links on articles is also estimated as a factor of relativity. For instance, assume that there is an article which is referred to from many other articles. This article would have a lot of short paths to many articles. This means that it has a strong relativity to many articles if we used only pf. However, this kind of articles must be considered as a general concept, and the importance of general concepts is not high in most cases. Therefore, we must consider the inversed backward link frequency ibf in addition to the two factors above.

4.2 Dual Binary Tree

The counting of all paths between all pairs of articles in a huge graph is a computational resource consuming work. Thus, making it efficient is a serious issue on Wikipedia mining. Using adjacency matrices and multiplication is not a clever idea because of the low scalability. Wikipedia has more than 1.3 million articles, thus we needs several terabytes just for storing data. Further, we need unimaginably much time to calculate the multiplication because the order is $O(N^3)$. However, a large number of elements in the adjacency matrix of a Web site are zero, thus effective compression data structures and analysis methods are the key to achieve high scalability on Wikipedia mining. Therefore, we propose an efficient data structure named "Dual binary tree" (DBT) and a multiplication algorithm for the DBT.

Since the adjacency matrix of a Web site link structure is a sparse matrix (almost all elements are zero), the DBT stores only the non-zero elements for data compression. Figure 2 shows the image of a DBT. The DBT consists of two types of binary trees; i-tree and j-tree. Each element in the i-tree corresponds to a row in the adjacency matrix and each i-tree element stores a pointer to the root of a j-tree. This means that the DBT consists of totally $N + 1$ (1 i-tree and

N j-trees) binary trees. The point is that operations for both getting and storing data are very fast because the number of steps is in both cases $O(logN)$.

Next, we define the multiplication algorithm for the DBT as follows:

Algorithm $MultiplyDBT(A)$
1 **for** $i \in$ i-$Tree$
2 **for** $j \in$ j-$Tree(i)$
3 **for** $k \in$ j-$Tree(j)$
4 $R_{i,k} := R_{i,k} + a_{j,k} \cdot a_{i,j}$;

The function j-$Tree(i)$ extracts all elements in the ith row of the adjacency matrix A. $a_{j,k}$ denotes the element in the jth row and kth column of the matrix. The first loop will be executed N times, but the numbers of cycles of the second and third loop depend on the average link number M. Thus the total number of steps is $O(M^2NlogN)$. Further, M is constantly 20 to 40 in Wikipedia in spite of the evolvement of the matrix size N. Finally, the result is stored in another DBT R.

We conducted a benchmark test for DBT and the multiplication algorithm compared with GNU Octave (with ATLAS library), one of the most effective numerical algebra implementations and the result shows the effectiveness of DBT for huge scale sparse matrix multiplication.

pfibf with DBT In this section, we describe the concrete flow of $pfibf$ calculation by using a DBT. Since $pfibf$ analyzes both forward and backward links of the articles, first we calculate A' by adding A and the transpose matrix A^T as follows:

$$A' = A + A^T. \tag{4}$$

By calculating the power of A', we can extract the number of paths for any pair of articles in n-hop range. An element $a'^n_{i,j}$ in matrix A'^n denotes the number of paths from article v_i to article v_j whose length is n. However, before calculating A'^n, each element in A should be replaced by the following formula to approximate ibf:

$$a'_{i,j} = a'_{i,j} \cdot log\frac{N}{|B_{v_j}|}. \tag{5}$$

$|B_{v_j}|$ denotes the number of backward links of article v_j. Finally, we can extract the $pfibf$ for any pair by adding the matrices A'^1, A'^2, ... , A'^n as follows:

$$pfibf(i,j) = \sum_{l=1}^{n} \frac{1}{d(n)} \cdot a'^l_{i,j}. \tag{6}$$

$d()$ denotes a monotonically increasing function such as a logarithm function which increases the value according to the length of path n.

4.3 FB Weighting

After a number of experiments to evaluate the accuracy of $pfibf$, we realized that the accuracy decreased in particular situations. Then, after having further experiments in order to detect the cause, we finally realized that the accuracy of general term analysis is worse than the accuracy of domain specific terms. General terms have the following characteristics:

- They have a lot of backward links,
- They are referred to from various topic-ranges,
- The content is trustful because it is usually edited by many authorities.

General terms, such as "United states," "Marriage" and "Horse," are referred to from various articles in various topic ranges. This means that the backward link analysis cannot be converged because the topic locality is weaker than in domain-specific terms such as "Microsoft" and "iPod." Although the backward link analysis is not convergent, the forward link analysis is effective because the contents are trustful and usually edited by many authorities.

In contrast to this, domain-specific terms have a much stronger topic locality. Although they have less links from other pages and the contents are sometimes not trustful, each link from other pages is topically related to the content. Therefore, we developed the "FB weighting" method which flexibly changes the weight of the forward link analysis and backward link analysis as follows:

$$W_b(|B_d|) = 0.5/(|B_d|^{\alpha}), \tag{7}$$

$$W_f(|B_d|) = 1 - W_b(|B_d|). \tag{8}$$

$|B_d|$ is the backward link number of article d. The constant α must be optimized according to the environment. After a number of experiments, an α value of about 0.05 was recognized to be suitable for the link structure of Wikipedia. The weight W_b is multiplied for each element on A and W_f for A^T as well. Thus formula (4) must be modified into the following formula (9):

$$A' = W_f \cdot A + W_b \cdot A^T. \tag{9}$$

5 Experiments

To evaluate the advantages of our approach, we conducted several experiments. In this section, we describe these experiments and discuss the results.

5.1 Overview

First of all, we constructed four thesauri from Wikipedia by using the four methods mentioned in this paper; TF-IDF, link cooccurrence analysis, $pfibf$ (2-hop) and $pfibf$ with FB (2-hop) weighting, in order to to evaluate the performance.

After that, we conducted two experiments to evaluate the accuracy of the constructed thesauri. In the first experiment, the accuracy of an association thesaurus extracted by each of the four methods was evaluated by the "WordSimilarity-353 test collection"[12] which has often been used in previous Wikipedia researches[4, 6]. The test collection contains 353 word pairs and these pairs have

been judged by 13-16 testers to produce a single relatedness score. For each method, we calculated the relatedness for each pair in this test collection. Then, we compared the extracted relatedness with the human judgements by using Spearman's rank correlation coefficient.

In the second experiment, the accuracy for each method was evaluated by CP (Concept Precision)[13]. We developed an evaluation interface which shows the top 30 associated terms for a query for each constructed thesaurus individually. Users evaluated whether the associated terms presented by the system are relevant or not by ranking them into 3 levels (1: irrelevant, 2: Moderate, 3: Relevant). CP is defined by the following formula:

$$CP = \frac{Number\ of\ retrieved\ relevant\ concepts}{Number\ of\ total\ retrieved\ concepts}. \tag{10}$$

"Number of retrieved relevant concepts" means the number of concepts that were scored 3 by users. We randomly gave 10 different queries and 120 associated terms (30 terms / method) from the queries, thus totally 300 term pairs were evaluated. 12 people participated in the experiment and on average 6 testers evaluated one set of pairs.

5.2 Result

Table 1 shows the experimental results.

Table 1. Performance and accuracy for thesaurus construction.

Methods	Time / Page	Experiment 1	Experiment 2 (CP)		
		(Spearman)	Top 10	Top 20	Top 30
TF-IDF	0.001 sec.	0.574	69.3%	66.9%	66.3%
Cooccurrence	0.001 sec.	0.538	65.7%	59.8%	55.8%
pfibf	0.30 sec.	0.677	76.3%	71.2%	68.3%
pfibf with FB	0.34 sec.	0.680	81.8%	75.3%	73.2%

First, the analysis time for TF-IDF and Cooccurrence was much shorter than that of pfibf because these are sequential approach while pfibf needs a huge scale matrix multiplication. pfibf (FB weighting as well) took totally 62 hours on a single workstation (Pentium 4 2.4 GHz) to extract 300 associated concepts for every concept (1.3 million concepts) in Wikipedia. Thus, by using a single workstation, we can extract the thesaurus once per several days. We believe that it is enough scalable for many applications such as IR systems with high-coverage for latest concepts.

However, the analysis time could be reduced by using several workstations because the DBT multiplication is suitable for parallel computing. We used 3 workstations to reconstruct the thesaurus and the result was exactly same but the analysis time become one third of that of a single workstation.

Regarding the two experiments, the results show that both our proposed methods achieved higher accuracy than the other two methods in both CP and Spearman's rank correlation coefficient. This means that the two factors of pfibf

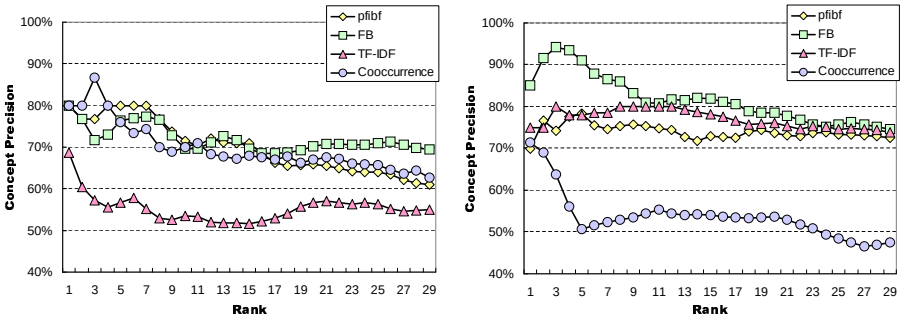

Fig. 3. CP (Concept Precision) of domain specific terms and general terms.

(number of paths and length of paths) are helpful in order to construct an accurate thesaurus. According to the CP, the accuracy of pfibf with FB weighting method is better than that of plain pfibf. However, by comparing Spearman's rank correlation coefficient, the accuracy of FB weighting is not much different from that of plain pfibf.

In order to make the effectiveness of FB weighting clear, we compared these four methods in detail. We separated the queries into 2 categories; domain specific terms and general terms. After that, we evaluated the methods in three cases; thesaurus construction for domain specific terms only, general terms only and all terms mixed.

Figure 3 (left) shows a comparison of CP for domain specific terms such as "Microsoft" and "PlayStation." It shows that the link cooccurrence analysis achieves a high precision for the top ranked terms. FB weighting proved to be less effective for the domain specific term analysis than the normal $pfibf$. As we mentioned before, the contents of domain specific terms (articles) are not refined enough compared with general terms, thus irrelevant links occur relatively often. We think that this is the reason why the CP of TF-IDF decreases so drastically for lower ranked pages.

Figure 3 (right) shows a comparison of the CP for general terms such as "Sport" and "Book." It shows that the CP is quite high for the top 10 terms extracted by FB weighting, and the CP decreases softly, but keeping a relatively high value even for pages with a very low rank. In contrast to this, the cooccurrence analysis was not effective for general terms except for the top terms. General terms cooccur with various terms in various topics because the topic locality of general terms is not strong. We think that this low topic locality is the cause of the accuracy problem.

Table 2 shows an example of an association thesaurus constructed by $pfibf$ with FB weighting.

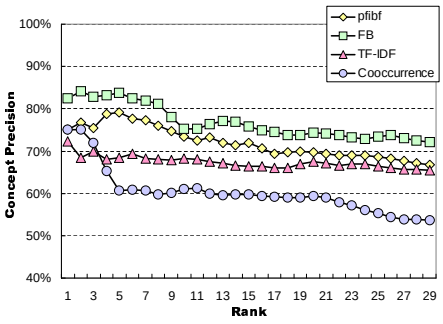

Fig. 4. Concept precision of all terms.

Table 2. Sample of queries and the extracted terms by pfibf with FB weighting.

Query	Extracted association terms		
Sports	Basketball	Baseball	Volleyball
Microsoft	MS Windows	OS	MS Office
Apple Comp.	Macintosh	Mac OS X	iPod
iPod	Apple Comp.	iPod mini	iTunes
Book	Library	Diamond Sutra	Printing
Google	Search engine	PageRank	Google search
Horse	Rodeo	Cowboy	Horse-racing
Film	Actor	Television	United States
DNA	RNA	Protein	Genetics
Canada	Ontario	Quebec	Toronto

6 Conclusion

Wikipedia, a very large scale Web-based encyclopedia, is an invaluable Web corpus for knowledge extraction. In this paper, we first unveiled the link structure of Wikipedia in detail to prove that it has a high potential for knowledge extraction. In the next step, we listed up the possible conventional methods that can be used for Wikipedia mining and proposed a link structure mining method $pfibf$, a scalable and high accuracy method for association thesaurus construction. After that, we applied FB weighting as an extension to avoid the accuracy problem on general terms. Finally we confirmed the notable advantage of our proposed methods in a number of experiments.

The constructed thesaurus (pfibf with FB weighting) is accessible under the following URL and it allows users to extract associated terms from any concept in Wikipedia.

`http://wikipedia-lab.org:8080/WikipediaThesaurusV2`

An association thesaurus is just a first step in our whole project; "Wikipedia mining." Our next step is another project called "Wikipedia Ontology;" a huge scale Web ontology which is automatically extracted by Wikipedia mining. The

purpose of this project is to extract not only term associations but also term relations such as "is-a" or "part-of."

We believe that Wiki-based knowledge management in enterprise environments will be popular in near future. This means that the application of our proposed methods, $pfibf$ and FB weighting, are not limited to Wikipedia. These methods also can be applied for extracting organization specific concepts.

7 Acknowledgment

This research was supported in part by Grant-in-Aid on Priority Areas (18049050), and by the Microsoft Research IJARC Core Project.

References

1. Giles, J.: Internet encyclopaedias go head to head. Nature **438** (2005) 900–901
2. Nakayama, K., Hara, T., Nishio, S.: A thesaurus construction method from large scale web dictionaries. In: Proc. of IEEE International Conference on Advanced Information Networking and Applications (AINA 2007). (2007) 932–939
3. Ruiz-Casado, M., Alfonseca, E., Castells, P.: Automatic assignment of wikipedia encyclopedic entries to wordnet synsets. In: Proc. of Advances in Web Intelligence Third International Atlantic Web IntelligenceConference (AWIC 2005). (2005) 380–386
4. Strube, M., Ponzetto, S.: WikiRelate! Computing semantic relatedness using Wikipedia. In: Proc. of National Conference on Artificial Intelligence (AAAI-06), Boston, Mass. (2006) 1419–1424
5. Milne, D., Medelyan, O., Witten, I.H.: Mining domain-specific thesauri from wikipedia: A case study. In: Proc. of ACM International Conference on Web Intelligence (WI'06). (2006) 442–448
6. Gabrilovich, E., Markovitch, S.: Computing semantic relatedness using wikipedia-based explicit semantic analysis. In: Proc. of International Joint Conference on Artificial Intelligence (IJCAI 2007). (2007) 1606–1611
7. Salton, G., McGill, M.: Introduction to Modern Information Retrieval. McGraw-Hill Book Company (1984)
8. Lawrence, P., Sergey, B., Rajeev, M., Terry, W.: The pagerank citation ranking: Bringing order to the web. Technical Report, Stanford Digital Library Technologies Project (1999)
9. Kleinberg, J.M.: Authoritative sources in a hyperlinked environment. Journal of the ACM (5) (1999) 604–632
10. Davison, B.D.: Topical locality in the web. Proc. of the ACM SIGIR (2000) 272–279
11. Schutze, H., Pedersen, J.O.: A cooccurrence-based thesaurus and two applications to information retrieval. International Journal of Information Processing and Management **33**(3) (1997) 307–318
12. Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., Ruppin, E.: Placing search in context: the concept revisited. ACM Trans. Inf. Syst. **20**(1) (2002) 116–131
13. Chen, H., Yim, T., Fye, D.: Automatic thesaurus generation for an electronic community system. Journal of the American Society for Information Science **46**(3) (1995) 175–193