

A Thesaurus Construction Method from Large Scale Web Dictionaries

Kotaro NAKAYAMA Takahiro HARA

Shojiro NISHIO

Dept. of Multimedia Eng., Graduate School of Information Science and Technology,
Osaka University, 1-5 Yamadaoka, Suita, Osaka 565-0871, Japan
{nakayama.kotaro, hara, nishio}@ist.osaka-u.ac.jp

Abstract

Web-based dictionaries, such as Wikipedia, have become dramatically popular among the internet users in past several years. The important characteristic of Web-based dictionary is not only the huge amount of articles, but also hyperlinks. Hyperlinks have various information more than just providing transfer function between pages. In this paper, we propose an efficient method to analyze the link structure of Web-based dictionaries to construct an association thesaurus. We have already applied it to Wikipedia, a huge scale Web-based dictionary which has a dense link structure, as a corpus. We developed a search engine for evaluation, then conducted a number of experiments to compare our method with other traditional methods such as co-occurrence analysis.

1. Introduction

A thesaurus is a set of terms and associations (relationships) among them. Although the effectiveness is widely proved by various research areas such as natural language processing (hence NLP) and Information retrieval, 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. WordNet[10] is a notable example made by a lot of contributors. WordNet contains over 80 thousand nouns and 90 thousands hierarchical relations. However, it is clear that thesaurus construction by human-effort is an annoying and time consuming work. Thousands of contributors have spend much time to construct high quality thesaurus dictionaries in the past. We should respect the contribution, however, it is difficult to maintain such huge thesauri, thus traditional thesauri do not support new concepts in most case. To solve this problem, 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.

Let us leave the topic on thesaurus construction and turn to Web-based dictionaries. Recently, Web-based dictionaries such as Wikipedia (Figure 1) have become dramatically popular among internet users as the WWW evolves. Some examples are shown as follows.

- Wikipedia
<http://www.wikipedia.org/>
- NetLingo
<http://www.netlingo.com/>
- FOLDOC (Free On-Line Dictionary Of Computing)
<http://foldoc.org/>
- Linktionary
<http://www.linktionary.com/>

It is needless to emphasis that one of the important characteristics of Web-based dictionaries are hyperlinks. Hyperlinks contain various information such as “Web locality.” Web locality, a law that the topics of connected pages by hyperlinks are more similar than pages not connected, has been proved to be true in many cases[6]. Of course, there is no doubt that Web locality also exists in Web-based dictionaries. Further, we believe that Web locality in Web-based dictionaries is stronger than on normal Web pages because the hyperlinks in Web-based dictionaries are explicit definitions of the concepts’ relations.

Concept identification based on URL is also a key feature on Web-based dictionaries. 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 term and there is no identification information whether

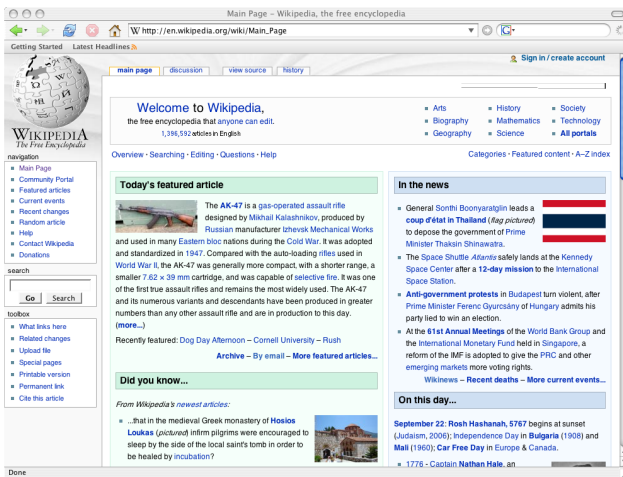


Figure 1. Wikipedia.

it is a fruit or a computer company. On the Web-based dictionaries, 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.

In our research, we focus on Wikipedia because it is an impressive Web content as a corpus for thesaurus construction. It is one of the biggest encyclopedias on the WWW, but the important characteristics are not limited to the huge amount of articles, its update frequency and dense link structure are also interesting characteristics.

Recently, Web structure mining for thesaurus construction is becoming a hot topic in the IR research area[3]. This approach has several advantages such as the reflection of latest term relations as the Web evolves. To construct a high quality thesaurus automatically, we believe that the corpus is as important as the analysis methods. What seems lacking in conventional methods, however, is the consideration of the Web corpus characteristics and optimization to the corpus. The analysis method should be optimized in both aspects; scalability and accuracy. Therefore, we propose an analysis method for Wikipedia mining which is both scalable and accurate in this paper.

The rest of this paper is organized as follows. In section 2, we introduce a number of researches on automated thesaurus construction in order to make our stance clear. In section 3, we describe the proposed method and show the result of our experiments in section 4. We draw a conclusion in section 5.

2. Related Works

A thesaurus is a data structure that defines semantic relatedness among words[14]. In this section, we introduce a number of studies for thesaurus construction which relate to

our research.

2.1. NLP based Thesaurus Construction

As we mentioned before, a large number of studies have been made on NLP-based thesaurus construction[13, 2, 16, 14, 5]. In fact, almost all popular traditional thesaurus construction methods are based on NLP. Traditional methods such as Co-occurrence analysis[14], n-gram analysis and tf-idf (term frequency-inverse document frequency) weighting[13] can be used for this purpose. These methods using NLP algorithms and tools like Brill's tagger[1], etc., are used in the preprocessing phase prior to term relativity analysis. A lot of outstanding studies have been done in order to improve the accuracy. However issues due to complexity of natural language such as ambiguous/synonym term problems still remain on NLP. This means there are many factors that complicate high-quality thesaurus construction on the preprocessing phases; stemming, morphological analysis, parsing, tagging, and so on.

2.2. Web Structure Mining

Web Mining is a wide research area, thus it is classified into several sub categories[4]; Web Contents Mining, Web Log Mining and Web Structure Mining. In this paper, we focus on Web structure mining. The main purpose of Web structure mining is to extract much information by analyzing the Web structure, mainly hyperlinks. Google's PageRank[9] and Kleinberg's HITS[8] are typical examples of Web structure mining. There are two type of links; "forward links" and "backward links" (See Figure 2). 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 PageRank and HITS algorithms, emphasize that the backward links are important in order to extract objective and trustful data.

In recent years novel methods for thesaurus construction based on Web structure mining are getting much attention. H. Chen et al.[3] have proposed an approach to automatically constructing domain-specific thesauri by the hyperlink structure analysis. The approach can be summarized as follows.

1. Selection of "high quality and representative websites" for a specific domain.
2. Extraction of the semantic relation among Web pages by using several heuristic rules.
3. Building of a website content structure for every selected website.
4. Merging of all obtained content structures by backward link text analysis.

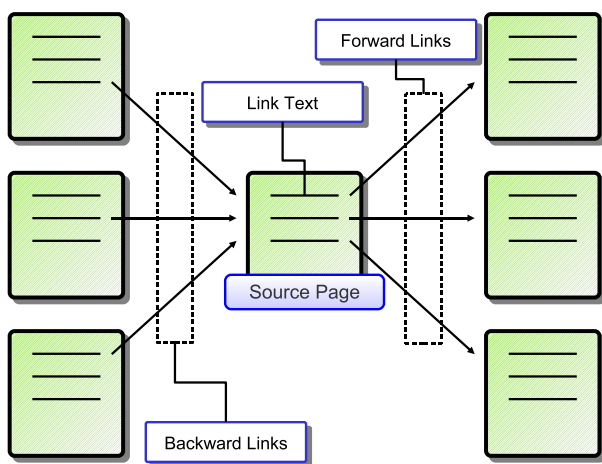


Figure 2. Types of link.

However, this approach is confronted by two difficulties; the NLP problem and the scalability. This algorithm still uses NLP tools to segment the link text while merging content structures of Web sites. Matching concepts by NLP is one of the most common causes of accuracy decrease due to the difficulties such as ambiguous/synonym term problems.

Scalability is also a serious problem. This algorithm constructs a lot of sub-trees to merge content structures, it then extracts term relations based on mutual information of a term-pair. If we apply this algorithm to very large Web-based dictionaries, the amount of sub-tree becomes large, thus it is estimated to be a time consuming analysis.

2.3. Wikipedia Mining

“Wikipedia mining” is a new research area which is recently addressed[11, 12, 15]. As we mentioned before, Wikipedia is an invaluable Web corpus for knowledge extraction. WikiRelate[15] 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 the 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 ($1.3 \text{ million} \times 1.3 \text{ million}$). 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.

We still need efficient methods suitable for Wikipedia

mining in both aspects; accuracy and scalability.

3. Efficient Link Structure Mining for Thesaurus Construction

We propose a scalable, efficient thesaurus construction method based on Wikipedia mining. In this section, we describe our approach in detail after describing the characteristic of Wikipedia.

3.1. Characteristics of Wikipedia

Wikipedia has become a huge phenomenon on the WWW. According to a statistics of Nature, Wikipedia is about as accurate in covering scientific topics as the Encyclopedia Britannica[7]. It covers various field concepts such as Arts, Geography, History, Science, Sports, Games and so on. It has more than 1.3 million articles (Sept. 2006) and it is becoming larger day by day. The number of entries reached 1 million (English only) in March 2006, but it was only half the size in March 2005.

The interesting characteristic of Wikipedia is not only the scale, but also the link structure. It has a very dense link structure. “Dense” means that it has a lot of “inner links,” links from pages in Wikipedia to pages in Wikipedia. This means articles are strongly connected by hyperlinks and there is no doubt that it is possible to extract important knowledge by analyzing the link structure.

Let us show some results of link structure analysis for Wikipedia. Figure 3 shows the distribution of backward links. It has Zipf distribution, containing a few nodes that have a very high degree and many with low degree. 196 pages have more than 10,000 backward links/page and 3,198 pages have more than 1,000 backward links/page, 67,515 pages have more than 100 backward links/page.

Totally 49,980,910 forward links (excluding redirect links) were extracted from 1,686,960 pages (excluding redirect, image, category pages). This means a page in Wikipedia has 29.62 forward links on average. Further, 2,531 pages have more than 500 forward links/page and 94,932 pages have more than 100 forward links/page. It can be concluded, from the statistics has been shown above, Wikipedia has a very dense link structure.

3.2. Basic strategy

A Web-based dictionary is a set of articles (concepts) and the hyperlinks among them, thus it can be expressed by a directed graph $G = \{V, E\}$ (V : set of articles, E : set of links). Let us consider how we can measure the relativity between any two articles. The relativity is assumed to be 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

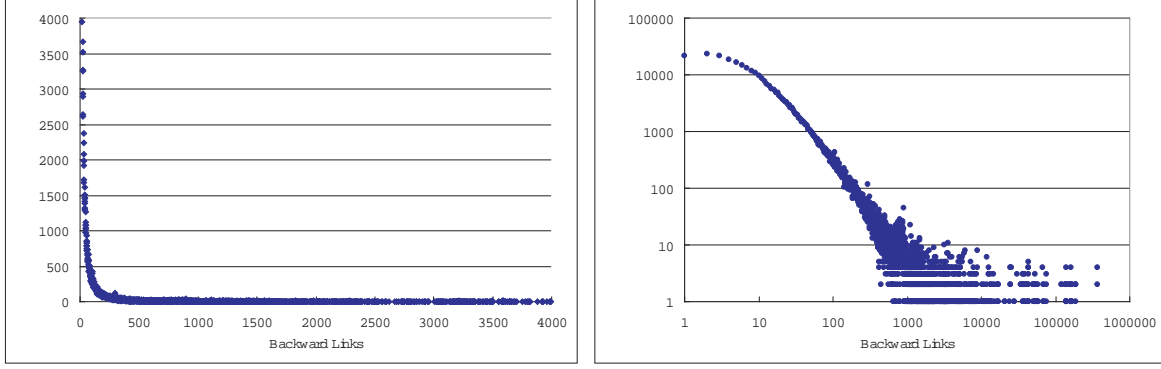


Figure 3. Distribution of amount of backward links.

The relativity becomes stronger if there are many paths (sharing many intermediate articles) between them. In addition, the relativity can be stronger according to the path length. In other words, if the articles are placed closely together in the graph G and sharing hyperlinks to articles, the relativity is estimated to be higher than further ones. Therefore, if all paths from v_i to v_j are given as $T = \{t_1, t_2, \dots, t_n\}$, the relativity lf (link frequency) between them can be expressed as follows:

$$lf(v_i, v_j) = \sum_{k=1}^n \frac{1}{d(|t_k|)}. \quad (1)$$

$d()$ denotes a function which increases the value according to the length of path t_k . A monotonically increasing function such as a logarithm function can be used for $d()$. In addition, the amount of links between individual articles is also estimated as a factor of relativity. For instance, assume that there is an article which has many links to other articles. Such 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 lf . However, this kind of articles must be considered as general concepts, and the importance of general concepts are not high in most cases. Therefore, we must consider the inversed backward link frequency ibf in addition to the two factors above. We therefore define an algorithm $lfibf$ to measure the relativity as follows:

$$lfibf(v_i, v_j) = lf(v_i, v_j) \cdot ibf(v_j). \quad (2)$$

$$ibf(v_j) = \log \frac{N}{bf(v_j)}. \quad (3)$$

N denotes the total number of articles and $bf(v_j)$ denotes the number of backward links of v_j . This means a page which shares hyperlinks with a specific page but not shares with other pages, has a high $lfibf$.

3.3. Mapping to natural language

The extracted association thesaurus can be used in various applications such as information retrieval, document categorization, and information filtering. However, most of these applications require the use of natural language for the user interface, thus mapping thesaurus entries to natural language is still an important issue. In this section, we consider ambiguous/synonym term detection problems and provide solutions by link structure mining.

First, we consider the ambiguous term detection problem. Ambiguous terms have various meanings, thus it is difficult to determine the meaning by NLP. For instance, “Apple” can mean a computer company or a fruit. However, since the text part of a backward link is a summary of the linked page in most cases, we realized that we can extract ambiguous term candidates by analyzing the backward links of a Web page. Based on this observation, we propose a backward link analysis method for ambiguous term detection. Figure 4 (left side) illustrates the idea of our method. It analyzes the backward links of all pages in the Web dictionary, then it extracts a set of pages which have exactly the same link text, these pages are the candidates for ambiguous terms. After the extraction, these ambiguous terms can be used for applications such as query-based web search engines. For a given natural language query string q , the concept specification method CS is defined as follows:

$$CS(v_i, q) = \frac{Cnt(B_{v_i}|q)}{\sum_{v_j \in V} Cnt(B_{v_j}|q)}. \quad (4)$$

$Cnt(B_{v_i}|q)$ is a function that returns the number of backward links of page v_i having the link text q . According to the result of several experiments, this strategy has proved to be effective. For instance, for the query “Apple” as q , the CS value scored 0.40 for apple as a fruit, 0.44 for Apple Computer as a computer company, and 0.09 for Apple Records as a record label. This means that the term “Apple” is widely used in roughly two meanings; a fruit

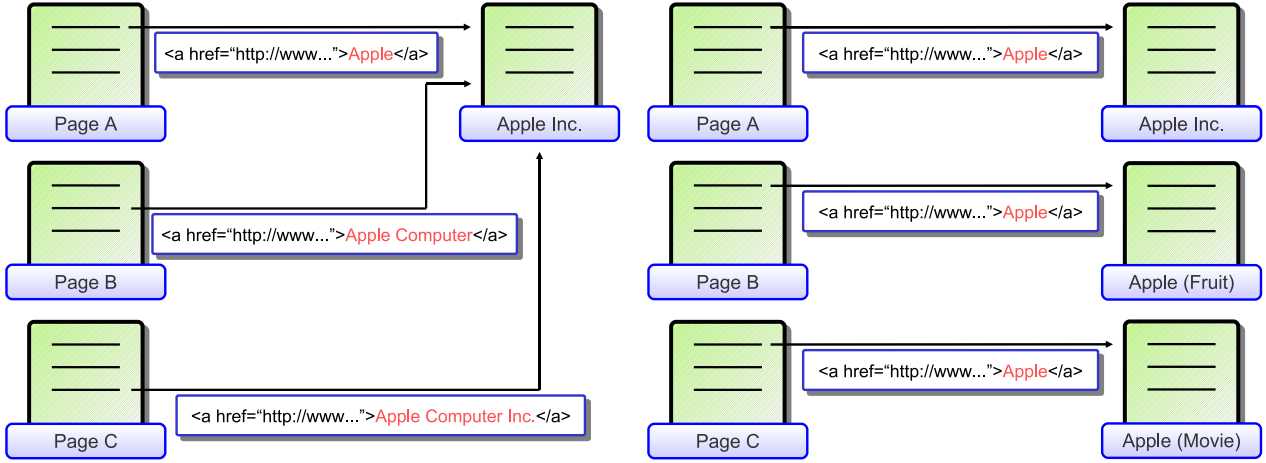


Figure 4. Ambiguous/synonym term detection by backward link analysis on Web dictionaries.

Table 1. Result of synonym extraction.

w	S_w
Apple Computer	176
Apple	462
Apple Computer Inc.	1
Apple Computers	2
...	...

and a computer company. One more example; “UFO” as q and “Unidentified_flying_object” as v_i , the CS value scored 0.65. This means that the query “UFO” is estimated to be equivalent to “Unidentified_flying_object” by 65% of possibility. By providing all possible candidates to the user and letting the user choose one of them, the meaning of a term can be specified.

We also discovered that we can extract synonyms by analyzing the backward links of an article. A synonym has one meaning but various expressions. For instance, “Apple Computer” is sometimes referred to as just “Apple.” Since backward links of a web page have a “variety of backward link texts,” this variety can be used to extract synonyms of a concept (article). Figure 4 (right side) shows an example of the variety of backward link texts. Table 1 shows the link texts of backward links of the page “Apple Computer.” This result shows that it can be written just “Apple”, but that is also sometimes written “Apple Computer, Inc.,” “Apple Computers,” etc.

We defined a synonym extraction algorithm for a concept v as follows:

Algorithm $GetSynonym(v)$

```

1  for  $b \in B_v$  do
2     $S_b := S_b + \frac{1}{|B_v|}$ ;
3  if  $S_b \geq \alpha$  then
4     $I_b := true$ ;
5  end;
6  end;

```

B_v denotes a set of backward links of concept v and b denotes a backward link text. Finally, I , a set of synonyms of concept v , is extracted. The threshold α depends on the dictionary we use as a corpus, thus we should optimize the value. 0.1 is empirically suitable for α in our experiments for Wikipedia mining. It means that the link text of b can be estimated to be a synonym of article v if the backward link text b occurs more than 10% in B .

4. Experiments

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

4.1. Overview

We developed a search engine (Fig. 5) that extracts a set of associated terms from a constructed thesaurus for any given query string. The top 30 associated terms for the query are shown for each thesaurus constructed by our approach, co-occurrence analysis and H. Chen’s method individually. Figure 6 shows the result page of the search engine. Users can evaluate whether the terms are relevant or not by ranking them into 5 levels (1: Definitely irrelevant,



Figure 5. Thesaurus search engine top page.

Table 2. Influence of distance to precision.

Hop	Top 10	Top 20	Top 30
1 hop	66.7%	64.2%	61.2%
2 hop	93.2%	86.2%	83.1%
3 hop	91.4%	89.4%	85.9%

2: Irrelevant, 3: Unknown, 4: Relevant, 5: Definitely relevant).

The results are evaluated by CP (Concept precision)[2]. CP is defined by the following formula:

$$CP = \frac{\text{Number of retrieved relevant concepts}}{\text{Number of total retrieved concepts}}. \quad (5)$$

“Number of retrieved relevant concepts” means the number of concepts which are scored 4 or 5 by user.

To prevent that the result depends on the users’ definition of the word “relevant,” we gave them the following instructions;

“If you can associate the extracted term from the query term, and you there exists a strong relation between them, it can be scored as a relevant term. In other words, if there is an explicit relation such as “is-a,” “part-of,” or “has-a” to the query term, these terms are relevant.”

4.2. Experiment 1

The first experiment was conducted to determine the optimal depth for link structure analysis in *lfibf*. In this experiment, we presented three sets of associated terms extracted by *lfibf* (1 hop, 2 hop, 3 hop analysis individually), then we let the user evaluate the relevancy of the terms by 5 levels. Totally 18 people participated in the experiment and

Table 3. Environment for performance evaluation.

Machine	Item	Spec.
Workstation for Analysis	CPU	Pentium4 3.2 GHz
	Main Memory	2 GB
	OS Implementation	Windows XP C#
DB Server	CPU	G4 1.42 GHz
	Main Memory	1 GB
	OS	Mac OS 10.4
	Database	MySQL 4.1

54 lists of terms are evaluated. Table 2 shows the result of the first experiment. “Top 10” means the concept precision on average for the top 10 terms in the ranking (sorted by the score of *lfibf*), “Top 20” and “Top 30” as well. The result shows that precision basically depends on the hop number for analysis. In fact, the result shows that the accuracy of the 2 hop analysis is dramatically better than the 1 hop analysis. However, the accuracy of the 3 hop analysis is not very different from the 2 hop analysis.

4.3. Experiment 2

The second experiment was conducted to measure the accuracy and the performance (analysis time) of the thesaurus construction. The experiment environment is shown in Table 3. We extracted a number of general terms as a set of query strings. Totally 17 people participated in the experiment. Table 4 shows the result and Table 5 shows an example of the thesaurus constructed by *lfibf* (2 hop analysis).

In H. Chen’s work, analysis depth of sub-trees was not considered, so we compared our approach to the 1 hop and 2 hop analysis of H. Chen’s algorithm. Since the 3 hop analysis in H. Chen’s algorithm did not finish in a practical time, we could not show the result for this case. Similar to our approach, the accuracy of the 2 hop analysis in H. Chen’s algorithm is dramatically better than that of the 1 hop analysis. Comparing the two approaches, our method achieves much higher accuracy than H. Chen’s algorithm. On the other hand, the experiment shows us that the analysis time in both approaches heavily depends on the analysis depth n . In summary, the accuracy of the 2 hop analysis of our method is as high as that of the 3 hop analysis because Wikipedia has a dense link structure even though the analysis time is dramatically faster than 3 hop analysis.

5. Conclusion

Wikipedia, a very large scale Web-based dictionary, is an impressive Web content as a corpus for thesaurus construc-

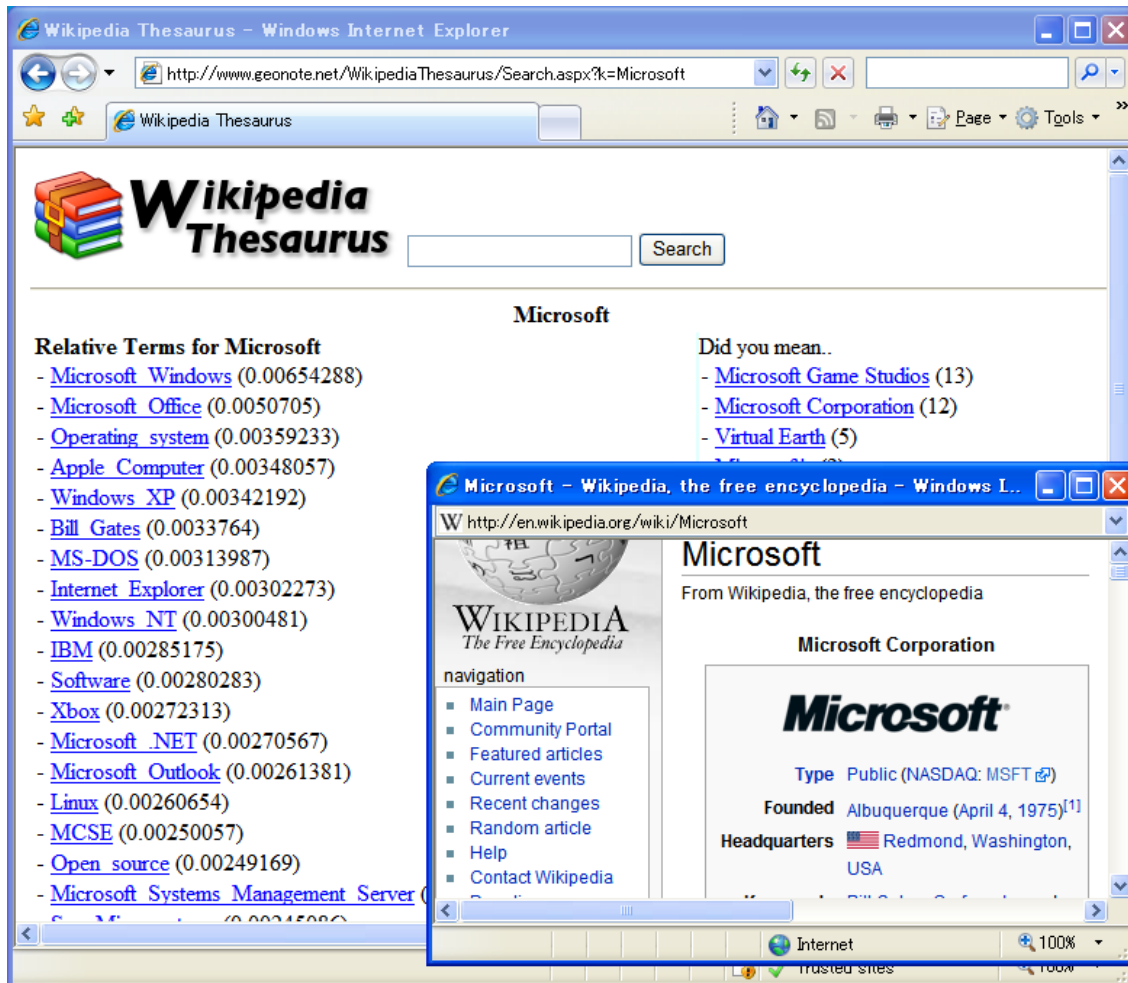


Figure 6. Search result page.

tion. In this paper, we proposed a Web thesaurus construction method based on Wikipedia mining. By analyzing 1.7 million concepts on Wikipedia, we constructed a very large scale association thesaurus which has more than 78 million associations. The accuracy is dramatically better than other methods based on NLP because we avoided the NLP problems by link structure mining for Web-based dictionaries.

The association thesaurus construction is just a first step in our whole project. Our next goal is another project called “Wikipedia Ontology;” a web-based ontology which is 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.”

6. Acknowledgment

This research was supported in part by Grant-in-Aid on Priority Areas (18049050), and by the 21st Century Center

of Excellence Program of the Ministry of Education, Culture, Sports, Science and Technology, Japan.

References

- [1] E. Brill. A simple rule-based part of speech tagger. *Proc. of Conference on Applied Computational Linguistics (ACL)*, pages 112–116, 1992.
- [2] H. Chen, T. Yim, and D. Fye. Automatic thesaurus generation for an electronic community system. *Journal of the American Society for Information Science*, 46(3):175–193, 1995.
- [3] Z. Chen, S. Liu, L. Wenyin, G. Pu, and W. Y. Ma. Building a web thesaurus from web link structure. *Proc. of the ACM SIGIR*, pages 48–55, 2003.
- [4] R. Cooley, B. Mobasher, and J. Srivastava. Web mining: Information and pattern discovery on the world wide web. *Proc. of the 9th IEEE International Conference on Tools with Artificial Intelligence*, pages 558–567, 1997.

Table 4. Comparison.

Methods	Top 10	Top 20	Top 30	analysis time / word
Co-occurrence	46.2%	35.4%	30.7%	0.34 sec.
Chen (1 hop)	39.3%	28.1%	22.4%	1.20 sec.
Chen (2 hop)	50.0%	50.9%	41.7%	121.34 sec.
Proposed method (1 hop)	66.7%	64.2%	61.2%	0.04 sec.
Proposed method (2 hop)	93.2%	86.2%	83.1%	4.00 sec.
Proposed method (3 hop)	91.4%	89.4%	85.9%	571.55 sec.

Table 5. Constructed thesaurus example.

Query	Associated terms extracted by our approach		
Sports	Basketball	Baseball	Volleyball
Microsoft	Microsoft Windows	Operating system	Microsoft Office
Video game	Nintendo	Japan	Video game developer
Apple Computer	Apple Macintosh	Mac OS X	iPod
Book	Library	Diamond Sutra	Printing
Rome	Italy	Pope	Public domain
Google	Search engine	PageRank	Google search
Football	FIFA	Penalty area	OFC Nations Cup
Watch	Clock	Chronometer	Jaeger-LeCoultre
Chicago	United States	Chicagoland	Illinois
New York City	United States	Race (U.S. Census)	Manhattan
Amazon River	Brazil	Peru	South America
Thomas Edison	Incandescent light bulb	Kinetoscope	Edison, New Jersey
Dog	The Intelligence of Dogs	Dog breed	American Kennel Club
Cat	Dog	Show cat	Cat breed
PDA	Mobile phone	EPOC (computing)	Pocket PC
Neuron	Glial cell	Action potential	Axon
Dictionary	Webster's Dictionary	Oxford English Dictionary	English language
Horse	Rodeo	Conditions races	Cowboy
Albert Einstein	Physics	Quantum mechanics	General relativity
Ontology	Semantic Web	Natural language processing	First-order logic
Earth	Moon	Sun	Earth's atmosphere
Imagine	Wonsaponatime	Live in New York City	John Lennon

- [5] C. J. Crouch. A cluster based approach to thesaurus construction. *Proc. of the ACM SIGIR*, pages 309–320, 1988.
- [6] B. D. Davison. Topical locality in the web. *Proc. of the ACM SIGIR*, pages 272–279, 2000.
- [7] J. Giles. Internet encyclopaedias go head to head. *Nature*, 438:900–901, 2005.
- [8] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, (5):604–632, 1999.
- [9] P. Lawrence, B. Sergey, M. Rajeev, and W. Terry. The pagerank citation ranking: Bringing order to the web. *Technical Report, Stanford Digital Library Technologies Project*, 1999.
- [10] G. A. Miller. Wordnet: A lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.
- [11] D. Milne, O. Medelyan, and I. H. Witten. Mining domain-specific thesauri from wikipedia: A case study. In *Proc. of ACM International Conference on Web Intelligence (WI'06)*, pages 442–448, 2006.
- [12] M. Ruiz-Casado, E. Alfonseca, and P. Castells. Automatic assignment of wikipedia encyclopedic entries to wordnet synsets. In *Proc. of Advances in Web Intelligence Third International Atlantic Web Intelligence Conference (AWIC 2005)*, pages 380–386, 2005.
- [13] G. Salton and M. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill Book Company, 1984.
- [14] H. Schutze and J. O. Pedersen. A cooccurrence-based thesaurus and two applications to information retrieval. *International Journal of Information Processing and Management*, 33(3):307–318, 1997.
- [15] M. Strube and S. Ponzetto. WikiRelate! Computing semantic relatedness using Wikipedia. In *Proc. of National Conference on Artificial Intelligence (AAAI-06)*, pages 1419–1424, Boston, Mass., July 2006.
- [16] Y. H. Tseng. Automatic thesaurus generation for chinese documents. *Journal of the American Society for Information Science and Technology*, 53(13):1130–1138, 2002.