Blind Evaluation for Thai Search Engines

Shisanu Tongchim*, Prapass Srichaivattana*, Virach Sornlertlamvanich*, Hitoshi Isahara†

Thai Computational Linguistics Laboratory
112 Paholyothin Road, Klong 1, Klong Luang
Pathumthani 12120, Thailand
*{shisanu,prapass,virach}@tcllab.org, †isahara@nict.go.jp

Abstract

This paper compares the effectiveness of two different Thai search engines by using a blind evaluation. The probabilistic-based dictionary-less search engine is evaluated against the traditional word-based indexing method. The web documents from 12 Thai newspaper web sites consisting of 83,453 documents are used as the test collection. The relevance judgment is conducted on the first five returned results from each system. The evaluation process is completely blind. That is, the retrieved documents from both systems are shown to the judges without any information about the search techniques. Statistical testing shows that the dictionary-less approach is better than the word-based indexing approach in terms of the number of found documents and the number of relevance documents.

1. Introduction

To date, the systematic and complete evaluation of different methods for implementing Thai search engines is not well investigated. The comparison between these methods is very important for improving the existing algorithms and developing new ones in the near future. In this study, we initiate a systematic evaluation between two different techniques that can be used to implement Thai search engines.

Developing a search engine for non-segmenting languages (e.g. Thai, Japanese and Chinese) is a challenging task. Traditional search engines rely on the construction of inverted index files. Some word segmentation algorithms are necessary in this state to specify word boundaries and determine the word entries. Then, the extracted word list is used to generate the indexes. Inevitably, the performance of search engines depends on the correctness of word segmentation modules. This kind of search engines relies on some dictionaries for word segmentation. Thus, the correctness and completeness of dictionaries is very important. In this research, these search engines will be referred to as the dictionary-based search engines.

Some researchers have proposed some alternative methods that do not rely on dictionaries (Sornlertlamvanich et al., 2003). By using a suffix array, the data is considered to be the sequence of characters and indexed character by character. The advantage of this indexing method is that it guarantees all search strings to be found, whereas the word-based indexing method depends on the word segmentation. Moreover, this indexing method can be applied to other languages without requiring any dictionary and language-specific knowledge. By using this indexing method, however, only some retrieved strings are meaningful. If the found pattern is a part of another word, that pattern is inseparable. As a result, it is not valid as a meaningful word. The validity of a word can be decided from its surroundings. If the word is closely connected to another word and unable to be separated from its context, it is likely to be a meaningless word. In contrast, the word is likely to be a meaningful word if it is loosely connected to another word. Some statistical measurements can be used to determine whether the word is separable from its surrounding context or not. One of measurements is the mutual information (MI). MI measures the degree of the co-occurrence of the query and its context. In conclusion, this search engine uses some techniques to identify the locations of queries, then some statistical measurements are used to determine whether the retrieved words are meaningful or not. In this research, these search engines will be referred to as the dictionary-less search engines.

In this paper, we conduct a blind evaluation between the dictionary-based and dictionary-less approaches. A web-based user interface is developed for judges. We provide natural language queries to judges. Then, each judge inputs desired keywords for each query to the user interface. The keywords will be sent to each search system. The results from both systems will be merged and presented in random order. After getting the results, our judges perform binary relevance judgments. Each document will be judged whether it is relevant or not.

The rest of this paper is organized as follows: Section 2 discusses related work. Section 3 provides the description of the techniques for implementing Thai search engines. Section 4 presents the evaluation process, experimental results and discussion. Finally, Section 5 concludes our work and discusses about possible future research.

2. Related Work

Leighton and Srivastava (1999) conducted a comparison among five commercial search engines in early 1997. They submitted 15 queries to search engines, and measured the precision on the first 20 returned results. However, they divided the first 20 links into three groups (namely, the first three links, the next seven links and the last 10 links), and assigned different weights to these groups. The findings showed that three search engines were superior to the other two.

In 1998, Gordon and Pathak (1999) compared eight search engines by using 33 topics from faculty members. All searches were performed by highly trained searchers. The assessment was done by the faculty members on the top 20 returned results from each search engine. The findings showed that absolute retrieval effectiveness was quite low.

Moreover, there were statistical differences among search engines for precision, but not the retrieval effectiveness.

Later, Hawking et al. (2001a) compared 20 search engines by using 54 topics originated by anonymous searchers. The top 20 results were judged. The findings showed that there was a significant difference in the performance of the search engines. They also did a comparison among 11 search engines using two different types of query (namely, online service queries and topic relevance queries) (Hawking et al., 2001b). They found a strong correlation between the performance results on both types of query.

The studies mentioned earlier compared performance of public search engines. The comparisons evaluated the performance of document retrieval algorithms, as well as the completeness of information collected by crawlers. In contrast to those studies, we decided not to conduct our experiment on public search engines. The reason is that we focus our attention on the document retrieval algorithms. Thus, we compare performance of set up machines with different retrieval algorithms instead. This is easier to control the experiment and examine the retrieval algorithms more closely.

3. Methods for Thai Document Retrieval

3.1. Dictionary-based Search Engine

The concept of typical dictionary-based search engines is shown in Fig. 1. At the first stage, some word segmentation algorithms are used to determine word boundaries. Then, the extracted words are used to construct the indexes. When searching is performed, the query is separated into words. These segmented words are used to search in the index file. Finally, the resulting documents are ranked according to some predefined scoring schemes.

As mentioned earlier, the performance of dictionary-based search engines is directly affected by the accuracy of word segmentation algorithms. Our previous work (Sornlertlamvanich et al., 2003) discussed about two possible errors affected by the accuracy of dictionary-based word segmentation modules. Assuming that a dictionary contains 6 words: **a, b, c, ac, bc** and **cb**.

- *Case 1: Incorrect word segmentation*

  The content of the document $\mathcal{A}$ is **abcbcb**. By using a word segmentation module, the content is separated into **a**|**bc**|**bc**|**b**. Assuming that the correct segmentation is **a**|**b**|**cb**|**cb**. If the query is **cb**, it cannot be found in the document $\mathcal{A}$ or if the query is **bc**, the document $\mathcal{A}$ will be incorrectly returned.

- *Case 2: Unregistered word problem*

  The content of document $\mathcal{A}$ is **abcdac**. By using the word segmentation module, the content is separated into **a**|**bc**|**d**|**ac**. Assuming that the correct segmentation is **a**|**b**|**cd**|**ac** and **cd** is an unregistered word to the word segmentation. If the query is **bc**, the result from this document will be incorrect or if the query is **cd**, it cannot be found in the document $\mathcal{A}$.

3.2. Dictionary-less Search Engine

The architecture of the dictionary-less search engine is illustrated in Fig. 2. It is composed of 3 major modules: (1) Data Indexing, (2) Searching and (3) Document Ranking.

3.3. Data Indexing

In typical search engines, web documents are separated into words to provide a word list for generating the indexes. In this approach, the data is considered to be the sequence of characters and indexed character by character. A suffix array is used to index the data. The advantage of this indexing method is that it guarantees all search strings to be found, whereas the word-based indexing method depends on the word segmentation. This indexing method can also be applied to other non-segmenting languages without requiring any dictionary and language-specific knowledge.

A suffix array for a string $\mathcal{S}$ of length n is an array of indexes or pointers, giving the the lexicographic ordering of the n suffixes of the string $\mathcal{S}$. If it is straightforwardly implemented, the size of the array will be equal to the size of input data. That is, all positions are indexed. However, some positions are inseparable. Thus, those positions can be skipped when indexing documents. By using a set of simple rules based on types of Thai characters, about a half of all positions can be skipped. An example set of grammar rules can be found from the article (Theeramunkong et al., 2000).

3.4. Searching

Based on a suffix array, it requires $\mathcal{O}(m \log n)$ for a straightforward implementation to access the string in the data, where m is a length of the search string. By using precomputed information about the longest common prefixes, the search time is improved to $\mathcal{O}(m + \log n)$ (Manber and Myers, 1993).

The use of a suffix array guarantees that all search strings will be found. However, only the meaningful strings are preferred. If the found pattern is a part of other word, that pattern is inseparable. As a result, it is not valid as a meaningful word.

For example, assuming that the search query is short and likely to be a part of other strings such as "ยา" (drug). Assuming that two strings are found: (1) "กินยา" (take a drug) and (2) "พัทยา" (Pattaya : name of a district in Thailand). The first string can be separated into two words: (1) "กิน" (take, eat) and (2) "ยา" (drug). Thus, the word "ยา" in the first string is a meaningful word. For the second string, the first part "พัท" is a meaningless word and is closely connected to the second part "ยา". Thus, the word "ยา" in the second string is meaningless since this string is inseparable.

From the example, the validity of a word can be decided from its surroundings. If the word is closely connected to another word and unable to be separated from its context, it is likely to be a meaningless word. In contrast, the word is likely to be a meaningful word if it is loosely connected to another word.

We use mutual information (MI) (Church and Hanks, 1989) to measure the degree of the co-occurrence of the query and its context. Let xy be a query, ab is the left context and cd is the right context of the string xy, the mutual

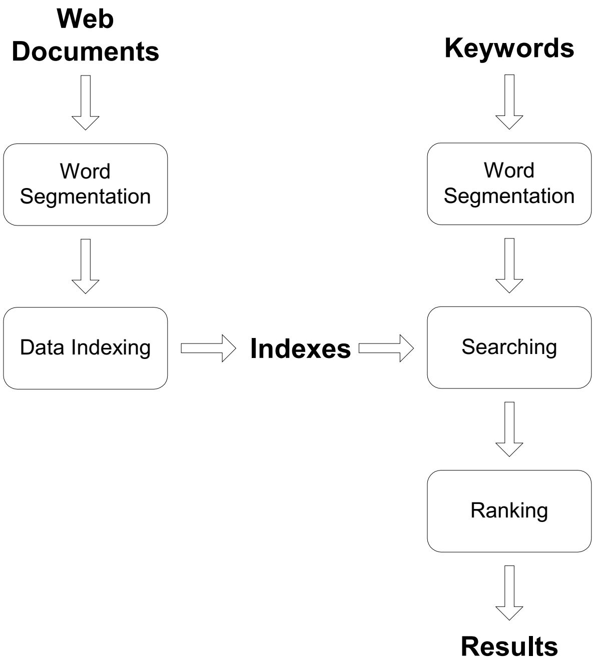

Figure 1: Typical Dictionary-based Search Engine

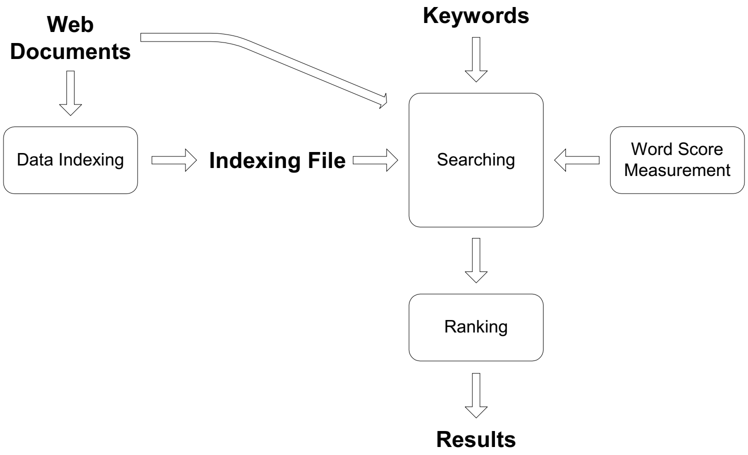

Figure 2: Dictionary-less Search Engine

information can be determined by the Equation 1-4.

$$MI_L(abxy) = \frac{p(abxy)}{p(ab)p(xy)} \quad (1)$$

$$MI_L(abxy) \approx \frac{Count(abxy)}{Count(ab)Count(xy)} \quad (2)$$

$$MI_R(xycd) = \frac{p(xycd)}{p(xy)p(cd)} \quad (3)$$

$$MI_R(xycd) \approx \frac{Count(xycd)}{Count(xy)Count(cd)} \quad (4)$$

If the MI value is high, xy is likely to be a part of the context. On the other hand, xy should be independent from the context if the MI value is low. We define the inverse of MI as the word score. The word score is calculated by the Equation 5-6.

$$wscore_L(xy|ab) = 1 - norm(MI_L(abxy)) \quad (5)$$

$$wscore_R(xy|cd) = 1 - norm(MI_R(xycd)) \quad (6)$$

The $norm()$ is the normalizing function which normalizes the argument from 0 to 1. At this point, the word score determines the probability of being a word of the string.

3.5. Document Ranking

The word score from the previous step is not only used to determine the word boundary, it is also used to rank the document. That is, the document with higher word score will attain high rank.

4. Experiment

The test corpus comes from 12 Thai newspaper web sites. The corpus consists of 83,453 web pages or 370

MByte (text only). For the dictionary-based search engine, we use DataparkSearch Engine[1] which is an open source web-based search engine. The dictionary-less search engine is based on the suffix array with the combination of the mutual information. We assigned 30 topics to 5 judges. Each topic was derived from news headlines. Note that each judge received the same set of topics.

We developed a web-based user interface for the evaluation. We asked the judges to read the assigned topics and determined some keywords for searching. Our user interface accepted keywords entered by the judges and then submitted these keywords to the search engines. For each query, the top 5 results from both systems were merged and presented to the judges. Since the evaluation is completely blind, no information about the search engines is shown. The relevance judgements are binary. Each result will be considered by the judges whether it is relevant to the topic or not. Note that each judge may use different keywords for each particular topic. Thus, the returned results for one topic may be different for each judge.

In the experiment, we consider only the top five returned results from each system. For a particular test topic, however, the number of returned results for one system may be less than five. Since our test domain is limited to a number of news documents, searching for relevance documents in such a small domain imposes a limitation in terms of the number of returned documents. By using inappropriate keywords, we found empirically that both systems may not return any relevance documents. Thus, we decided not to use some precision-oriented measures like some studies (Gordon and Pathak, 1999; Hawking et al., 2001a). We compared two different document retrieval algorithms by using the number of returned results and relevance results. Since both measurements based on calculation at cutoff 5, the maximum number of both measurements is 25. The results are shown in Table 1.

To make the results clearer to see, the number of returned results and relevance results between two methods are compared by using the Wilcoxon signed-ranks test (Wilcoxon, 1945) at significance $\alpha = 0.005$. Statistical testing shows a significant difference between two methods for both measurements. The dictionary-less approach can retrieve more documents than those of the dictionary-based approach. In terms of the number of relevance documents, the dictionary-less approach is also better than the dictionary-based approach.

However, the number of relevance documents may not provide an accurate view of the performance in terms of the accuracy of returned document. In some topics, we cannot measure the number of relevance documents since no document can be found by the system. Moreover, the results also show that the numbers of relevance documents of the dictionary-based approach may be higher than those of the dictionary-less one even the numbers of returned documents are lower. This observation is confirmed by statistic testing. We calculated a precision-oriented measurement by using the proportion of returned documents which are relevant, and defined the value of this measurement as zero when no document is found. By using the Wilcoxon signed-ranks test at significance $\alpha = 0.005$, there is no significant difference between two methods. This can be interpreted as the proportions of retrieved documents which are relevant between two methods are not significant different. In general, the dictionary-less approach can find more documents, and thus the number of relevance documents.

There is one topic that the dictionary-based approach returns more documents than those of the dictionary-less approach. From the detailed analysis, one user used some long keywords in that topic. The dictionary-less approach searches directly from the input keywords without segmenting keywords beforehand. The algorithm performs exact string matching for given keywords. If the given keywords is too specific, it may not find any relevance documents. This is a limitation of the dictionary-less approach.

5. Conclusions and Future Work

This paper compares the dictionary-less search engine against the dictionary-based one using a blind evaluation method. Statistical testing shows differences between both document retrieval algorithms on the number of returned documents and the relevance documents. However, the proportions of relevance documents to retrieved documents are not significant different.

In the future, we plan to extend our comparison to other search approaches that can be used to implement Thai search engines. We also plan to include other measurements (e.g., search time, memory usage) in the performance comparison.

6. References

K.W. Church and P. Hanks. 1989. Word association norms, mutual information and lexicography. In *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics*, pages 76–83.

M. Gordon and P. Pathak. 1999. Finding information on the world wide web: The retrieval effectiveness of search engines. *Information Processing and Management*, 35(2):141–180.

D. Hawking, N. Craswell, P. Bailey, and K. Griffiths. 2001a. Measuring search engine quality. *Information Retrieval*, 4(1):33–59.

D. Hawking, N. Craswell, and K. Griffiths. 2001b. Which search engine is best at finding online services? In *Poster Proceedings of the Tenth International World Wide Web Conference*.

H.V. Leighton and J. Srivastava. 1999. First 20 precision among world web search services (search engines). *Journal of the American Society for Information Science*, 50(10):870–881.

U. Manber and E.W. Myers. 1993. Suffix arrays: A new method for on-line string searches. *SIAM Journal on Computing*, 22(5):935–948.

V. Sornlertlamvanich, P. Tarsaku, P. Srichaivattana, T. Charoenporn, and H. Isahara. 2003. Dictionary-less search engine for the collaborative database. In *Proceedings of the Third International Symposium on Communications and Information Technologies*, volume 1, pages 177–182, September.

[1] http://www.dataparksearch.org/

T. Theeramunkong, V. Sornlertlamvanich, T. Tanhermhong, and W. Chinnan. 2000. Character cluster based thai information retrieval. In *Proceedings of the Fifth International Workshop on Information Retrieval with Asian Languages*, pages 75–80, September-October.

F. Wilcoxon. 1945. Individual comparisons by ranking methods. *Biometrics*, 1:80–83.

Topic	Number of Returned Results		Number of Relevance Results	
	Dictionary-based	Dictionary-less	Dictionary-based	Dictionary-less
1	17	22	6	6
2	23	25	18	17
3	22	23	16	9
4	7	19	5	12
5	20	25	9	18
6	19	25	17	17
7	0	22	0	18
8	20	25	19	11
9	14	25	6	19
10	20	23	15	10
11	0	25	0	11
12	18	25	6	21
13	25	20	10	13
14	20	25	18	11
15	16	25	14	15
16	15	21	3	10
17	20	22	9	7
18	25	25	9	17
19	16	25	7	18
20	17	21	12	10
21	25	25	13	21
22	25	25	9	12
23	21	21	13	15
24	25	25	14	17
25	0	25	0	21
26	1	23	0	16
27	9	25	1	15
28	15	25	8	15
29	23	25	11	18
30	19	25	9	16

Table 1: Number of returned results and relevance results for the test topics