THAI TEXT MINING TO SUPPORT WEB SEARCH FOR E-COMMERCE

Todsanai Chumwatana, Kok Wai Wong, and Hong Xie

School of Information Technology
Murdoch University
South St, Murdoch
Western Australia 6150

ABSTRACT

E-commerce has grown rapidly since the internet or other computer networks become popular. Today, one can observe a wide variety of products and services that can be supported by e-commerce. For e-commerce to be successful, numerous resources and technology are used. For example, users may spend huge amount time to find the information they want on the Internet. Huge storage space may be used to index or store the e-commerce web pages and catalogues. As e-commerce becomes entrenched in the business world, e-commerce web sites are becoming more and more complex. One of the challenges develop efficient methods for constructing the index for web search that apply not only to European languages but also to Asian languages due to the difference language characteristics around the world. This paper aims to employ the frequent max substring mining for indexing the e-commerce web contents. We use the improved frequent suffix trie or FST properties to reduce the possibility of constructing full index for the web search. The assumption made here is basically we are interested only in the longest substrings that occur frequently. This assumption works well with Thai language which we are focusing on in this paper. This improve technique requires low storage space to keep only part of the longest frequent substrings. At the same time, the improved technique enables efficient retrieval for customers to search the information of several Thai e-commerce web pages.

Index Terms— *Frequent max substring, the longest substring, e-commerce, text mining and Frequent suffix trie.*

1. INTRODUCTION

The rapid growth of the e-commerce has exceeded all expectations in the past few years. Since the Internet has become popular worldwide in the middle of 1990s. E-commerce has been expanding at a very fast pace, this makes e-commerce the largest publicly accessible electronic markets in the world. This expansion is largely driven by the technological progress of the Internet. And by the end of 2000, many American business companies offered their services through the World Wide Web. Since then people began to associate the word "e-commerce" with the ability of purchasing various goods through the Internet using secure protocols and electronic payment services. However, not only in America, e-commerce is also popular in Asia, Europe and Latin America. It is bringing enormous change in business firms, markets, and consumer behavior. Subsequently, e-commerce has become more complicated due to a large amount of e-business over the world, a variety of the web pages with many unique characteristics and in different languages, making e-commerce a very difficult and challenging task to attract potential customers. Therefore, web search is regarded as an important technique to attract consumers to the e-commerce sites. Web search is a technique that helps consumers to find the required information on World Wide Web. Web search has a strong relationship with information retrieval (IR). IR is a technique that helps users to find the required information from a large collection of web pages. The basic method of retrieval is to use the index to retrieve web pages that are likely to be relevant to the consumer's query. The index is also used to compute their relevance scores for ranking the retrieved web pages. For most cases, web pages are indexed by the indexer for efficient retrieval.

2. THAI TEXT MINING

Many IR techniques are designed for English language and other European languages, where the word boundary and characteristic are clearly defined. It is easy to classify terms and to construct the inverted index for English language web pages. E-commerce has becoming popular mainly because its technologies can be used to reduce supply chain costs, increase production efficiency, and tighten the relationship with customers. This makes the success of the e-commerce rippling around the world. There are many non-European language e-commerce web pages over the World Wide Web such as Thai, Chinese, Japanese, Korean and others. These languages are quite different to the English language because the characteristic of these languages is a string of symbols without explicit word boundary delimiter. Unlike English language, the structure of these languages is highly

ambiguous and is not delimited by spaces or any special characters. Consequently, when the word boundary is not clear, it can be difficult for term classification and the construction of index to allow efficient retrieval. In Thai language, Thai word segmentation is one of many techniques that used to support constructing index for Thai web search. Thai word segmentation can be divided into two major approaches, as Dictionary-based [26], [24], [25], [14], [16], [18] [4], [11], [10] and Non Dictionary-based or rule-based approaches [23], [17], [8], [7], [27], [13], [19], [20]. In addition, full text scanning [2], [15], [5], [1], [21], [12] in term of pattern matching could be used as one option to cope with this language by scanning through all documents sequentially. The objective for such method is to find the documents that contain the user query terms using matching string but the search time could be quite long depending on the documents. However, the limitations of Thai word segmentation are it can only be used for Thai language, and its storage structure can only be used for collecting Thai vocabulary. In order to solve these shortcomings, we propose an extensive algorithm for mining a set of frequent max substring patterns called FM from web pages where the word boundary is not explicitly defined to make terms classification. FM also keeps the frequency and a list of positions of terms to enable the computation of scores for ranking web pages. The access and generation of FM are speeded up by keeping positions to all occurrences of each frequent max substring pattern.

3. FREQUENT MAX SUBSTRING

In this paper, we aim to employ frequent max substring mining technique to classify terms from the contents of web pages where the word boundary and characteristic are not clearly defined and to allow the construction of the index file using frequent suffix trie or FST data structure. The methodology is to construct only the subtrees of suffix trie that correspond to the frequent max substrings of the contents. Vilo's algorithm [9] presented an efficient and scalable technique to discover frequent substrings, but it uses a large amount of storage space to keep all frequent substrings. In this paper, we use the frequent max substring mining for indexing [22] to construct web page content for Thai language. We construct only subtrees that correspond to the frequent max substrings which contain all frequent substrings. This leads more efficient algorithm and less storage requirement for storing and mining all frequent substrings.

Definition of Frequent suffix trie or FST structure

A set of all suffixes of an n-length string T or $S[s_i..s_n]$; where $1 \le i \le n$, is a set of substrings of string T that begins at position i and ends at position n [6].

FST structure of *n-length* string T is tree structure that represents all suffixes of string T starts with root node and ends with n leaf nodes. Also '$' is appended at the end of string T. $, the terminating symbol, is added to show the end of string T and to make all suffixes of string T different from each other. Therefore, all suffixes of string T contain $ at the n different ends of the FST structure.

FST structure also shows all substring patterns with their frequencies and positions of any substring patterns of string T.

Edge is a symbol or a character that is an element of the character set. Each edge starts with the same character, and then an extra character is added at each edge.

A node is used to represent a substring pattern with frequency and list of positions (or .pos). The position is the end position of each substring pattern of string T. Each depth of node leads to increased length of substring patterns. All leaf nodes keep suffixes with their frequencies and positions of suffixes.

4. CASE STUDY

1. Traditional algorithm

Suffix trie structure [3] is the traditional method to enumerate all possible substring patterns from the string but it enumerates only substring patterns without their frequencies. Also, pattern trie[9] is a trie structure that keeps the starting position of substring patterns without their frequencies. Therefore, this paper intends to use frequent suffix trie or FST and frequent max substring mining technique [22] to mine the long substring patterns that are likely to be essential of the web page contents with their frequencies and list of positions. This is mainly based on the assumption that the substrings that occur frequently in the web page contents should be the important keys of web pages.

The next example shows FST structure representing all substring patterns of Thai language using traditional algorithm to enumerate all suffixes of the string.

Let string T = ก า ร ป ร ะ ก อ บ ก า ร

1) Append $ to the string and define the position of each character in the string.

String T:	ก	า	ร	ป	ร	ะ	ก	อ	บ	ก	า	ร	$
Position (pos):	1	2	3	4	5	6	7	8	9	10	11	12	13

2) Enumerate all suffixes of the string

3) All suffixes are used to create the FST structure, as shown figure 1.

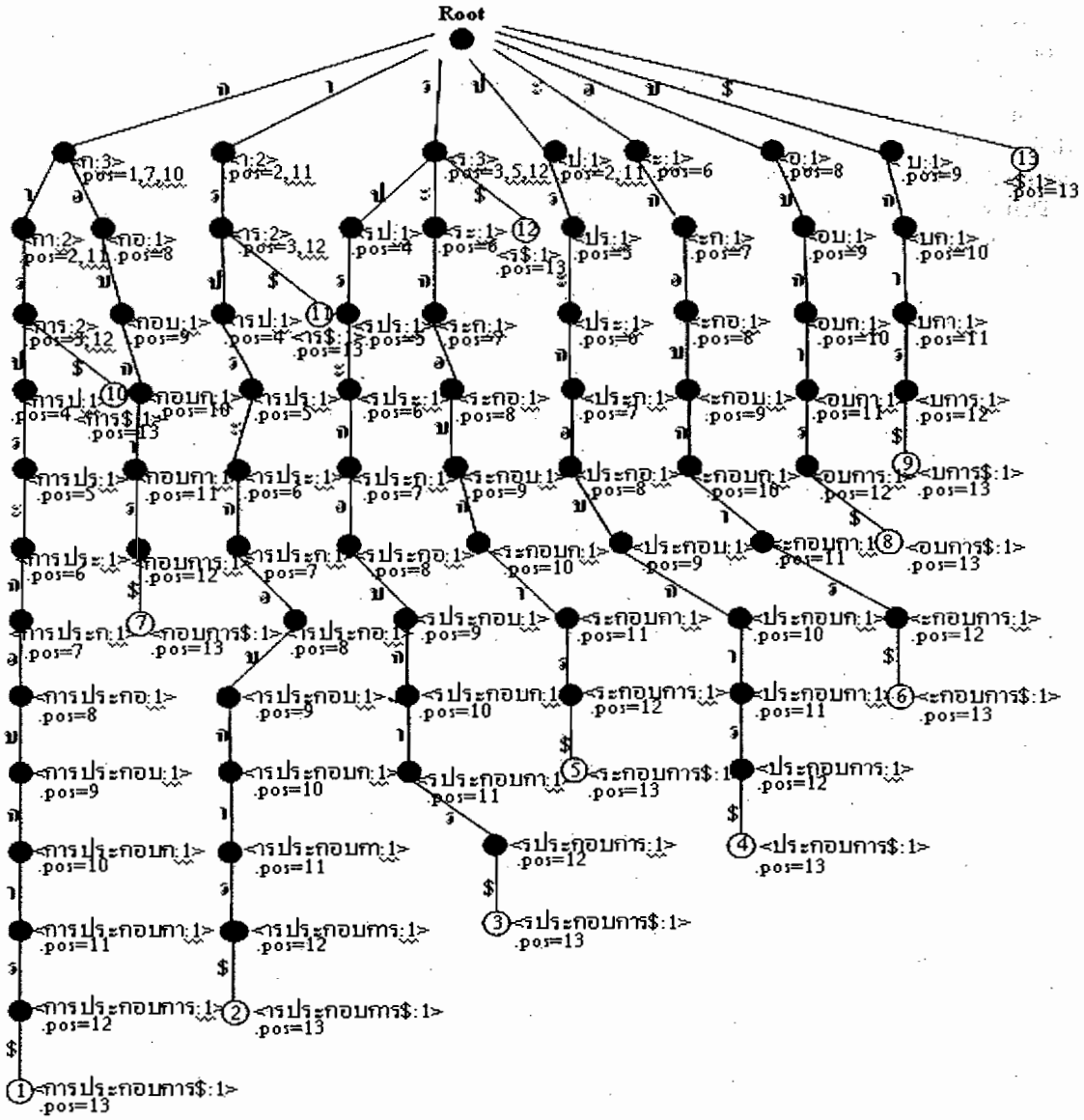

Figure 1 shows the FST structure for string T = ก า ร ป ร ะ ก อ บ ก า ร $

From figure 1, frequent max substrings are mined by the following steps.

1. Enumerate all substring patterns of string T using the FST structure.
2. Mine the set of frequent substrings whose frequency is not less than the defined frequency from the FST structure.
3. Mine the frequent max substring patterns from frequent substrings set.

From the algorithm, frequent max substrings mining must firstly enumerate all substrings, follow by mining the set of frequent substring and frequent max substring. Therefore, a large memory has to be used to keep all substrings. Thus, a resolution is a reduction of the number of substrings using two reduction rules; (1) reduction path rules using defined frequency to check mining termination, (2) reduction path rules using super-substring definition.

Efficient algorithm [22]

This algorithm uses the two reduction path rules to reduce the number of mining substring patterns. In this technique, the algorithm also uses heap data structure to support computation. Mining frequent max substrings using efficient algorithm is shown as the following steps.

Let string T = ก า ร ป ร ะ ก อ บ ก า ร $

and defined frequency = 2

1. Enumerate the *1*-length substring patterns with their frequencies, and then select frequent substring patterns

Substring patterns, frequency and position transaction (.pos) are kept in Min Heap structure sorted by order occurring in contents. The prior substring patterns can be more frequent max substring than later substring patterns.

2. Enumerate the child substring patterns of a substring pattern in Min Heap to process, and select only frequent child substring patterns. Min Heap structure will be then updated by using deletion rule. The deletion rule will check which child substring patterns are super-substrings of substring patterns in the Min Heap structure. If the frequency of substring pattern in Min Heap minus the frequency of super-substring pattern is less than defined frequency, the substring pattern will be deleted from the Min Heap structure and frequent child substring patterns are inserted in Min Heap by considering two rules; (1) substring pattern will be inserted to Min Heap structure ordered by the occurring position on string *T*, (2) if the first position of the substring pattern is equal to the first position of an existing substring in Min Heap, a substring pattern is inserted in the last position in the same group. The processed substring patterns are deleted from Min Heap. The other substring patterns will be processed until Min Heap is empty.
3. Mine frequent max substring by selecting substring patterns having no super-substring patterns from set of substring patterns in Min Heap.

From above steps, the FST structure is shown figure 2

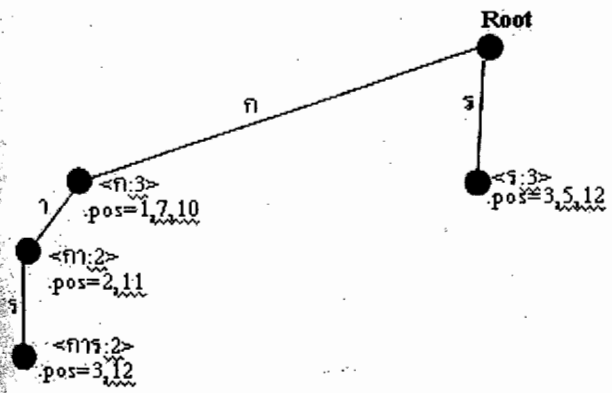

Figure 2 shows the FST structure using the efficiency algorithm.

Figure 2 shows the FST structure that contains some frequent substrings and all frequent max substrings .

5. CONCLUSION

This paper proposes using FM mining algorithm to support index construction in web search for e-commerce. While e-commerce is widely adopted the web is becoming the biggest and most widely known information source that is easily searchable and accessible in present society. Web has also become an important channel for businesses. Customers can buy almost anything on the internet from any part of the world. Our algorithm will be able to construct the index using less storage space to support information expansion. The algorithm can be used for Thai language, and is also able to be used with any language that has similar structure with Thai language and for other languages that are highly ambiguous and are not delimited by spaces or any special characters. Our algorithm used an improved suffix trie structure, called FST structure. FST structure is exploited to reduce the number of computations using two rules. As observed from the results, this algorithm gives less number of substring patterns than the traditional algorithm and Vilo's algorithm. This has improved over the original algorithm in term of memory storage space, computing efficiency and scalability.

6. REFERENCES

[1] A.V. Aho and M.J. Corasick. Fast pattern matching: an aid to bibliographic search. CACM, 18(6):333-340, June 1975.

[2] D.E. Knuth, J.H. Morris, and V.R. Pratt. Fast pattern matching in strings. SIAM J. Comput, 6(2):323-350, June 1977.

[3] D. Gusfield, Algorithms on Strings, Trees and Sequences Computer Science and Computational Biology. Cambridge: Cambridge University Press, 1997.

[4] Dister Anne 1998-1999. Développer des grammaires locales de levée d.ambiguités pour INTEX. In: FAIRON Cédrick (ed.) 1998-1999, pp. 233-247

[5] D.M. Sunday. A very fast substring search algorithm. Comm. of ACM (CACM),33(8):132-142, August 1990.

[6] D. R. Morrison, "PATRICIA - practical algorithm to retrieve information coded in alphanumeric," *Jrnl. A.C.M.*, pp. 15(4):514-534, 1968.

[7] D. Sawamibhadhi, "Implementation of Thai Grammar Analysis Software under UNIX system (in Thai)", Thammasart Univ., 1990.

[8] D. Sintupunpratum and C. Bandhitanont, "Thai Word Processing (in Thai)", Proc. of the second Symposium on Natural Language Processing in Thailand, pp. 322-376, March 1993.

[9] J. Vilo, "Discovering Frequent Patterns from Strings: Department of Computer Science. University of Helsinki, Finland," *Technical Report C-1998-9*, p. 20, May 1998.

[10] Krit Kosawat, Méthodes de segmentation et d'analyse automatique de textes thai Automated methods of segmentation and analysis of Thai texts, Thése pour obtenir le grade de

Docteur de l.Université de Marne-La-Vallée, le 8 septembre 2003.

[11] Kosawat Krit 2000. Procédure de reconnaissance des mots et des phrases thai. In: DISTER Anne (ed.) 2000, pp. 241-255.

[12] L.A. Hollaar, K.F. Smith, W.H. Chow, P.A. Emrath, and R.L. Haskin. Architecture and operation of a large, full-text information-retrieval system. In D.K. Hsiao, editor, Advanced Database Machine Architecture, pages 256-299. Prentice-Hall, Englewood Cliffs, New Jersey, 1983.

[13] Meknavin Surapant, CHAROENPORNSAWAT Paisarn, KIJSIRIKUL Boonserm 1997. Featurebased Thai Word Segmentation. Proceedings of the Natural Language ProcessingPacific Rim Symposium 1997 (NLPRS.97), 2-4 December 1997. Phuket, pp. 41-46.

[14] Raruenron Samphan 1991. Dictionary-based Thai Word Separation. [in Thai] Senior Project Report. Department of Computer Engineering, Chulalongkorn University, Bangkok.

[15] R.S. Boyer and J.S. Moore. A fast string searching algorithm. CACM, 20(10):762-772, October 1997.

[16] Sawamipak Duangkaew 1990. Construction of Thai Syntax Analysing Software under UNIX. [in Thai] Thammasart University Press, Bangkok.

[17] S. Charnyapornpong, "A Thai Syllable Separation Algorithm," M.Eng. Thesis, Asian Institute of Technology, Aug. 1983.

[18] Sorlertlamvanich Virach 1993. Word Segmentation for Thai in Machine Translation System. [in Thai] Machine Translation, pp. 50-56. NECTEC, Bangkok.

[19] Sorlertlamvanich Virach, Charoenporn Thatsanee, ISAHARA Hitoshi 1997. ORCHID Thai Part-Of-Speech Tagged Corpus. Technical Report: TR-NECTEC-1997-001. NECTEC, Bangkok.

[20] Sorlertlamvanich Virach, Potipiti Tanapong , Charoenporn Thatsanee, (2000). "Automatic Corpus-Based Thai Word Extraction with the C4.5 Learning Algorithm", vol.2, Jul 2000, pp. 802-807.

[21] Sun Wu and Udi Manber. Agrep – a fast approximate pattern searching tool. In USENIX Conference, January 1992.

[22] T. Chumwatana, , Kok Wai Wong and Hong Xie, Frequent max substring mining for indexing. International Journal of Computer Science and System Analysis (IJCSSA), "in progress" 2008.

[23] Thawaranon Kobkool 1978. Spacing in the Thai Writing. [in Thai] Master Thesis of Arts. Department of Thai Language, Chulalongkorn University, Bangkok.

[24] Varakulsiripunth Ruttikorn, Ngamwiwit Jongkol, Junwun Somsak, Chiwattayakul Suthatip, Thipchaksurat Sakchai 1989. Word Segmentation in Thai Sentence by Longest Word Mapping. [in Thai] In: Sornlertlamvanich Virach (ed.) 1995, pp. 279-290.

[25] V. Sornler tlamvanich, "Thai Word Segmentation in Language Translation Syst em," Computerized Language Translation (in Thai), p. 50-55, 1993.

[26] Y. Poovor awan and V. Imarom, "Dictionary-based Thai Syllable Segmentation (in Thai)," 9th Electrical Engineering Conference, 1986.

[27] Y. Thairatananond, "Towords the design of a Thai text syllable analyzer". Master Thesis Asian Institute of Technology.