Proceedings of the Sixth International Workshop on Information
Retrieval with Asian Languages, July 2003, pp. 9-16.

A Practical Text Summarizer by Paragraph Extraction for Thai

Chuleerat Jaruskulchai and Canasai Kruengkrai
Intelligent Information Retrieval and Database Laboratory
Department of Computer Science, Faculty of Science
Kasetsart University, Bangkok, Thailand
fscichj,g4364115@ku.ac.th

Abstract

In this paper, we propose a practical approach for extracting the most relevant paragraphs from the original document to form a summary for Thai text. The idea of our approach is to exploit both the local and global properties of paragraphs. The local property can be considered as clusters of significant words within each paragraph, while the global property can be though of as relations of all paragraphs in a document. These two properties are combined for ranking and extracting summaries. Experimental results on real-world data sets are encouraging.

1 Introduction

The growth of electronic texts is becoming increasingly common. Newspapers or magazines tend to be available on the World-Wide Web. Summarizing these texts can help users access to the information content more quickly. However, doing this task by humans is costly and time-consuming. Automatic text summarization is a solution for dealing with this problem.

Automatic text summarization can be broadly classified into two approaches: abstraction and extraction. In contrast to abstraction that requires using heavy machinery from natural language processing (NLP), including grammars and lexicons for parsing and generation (Hahn and Mani, 2000), extraction can be easily viewed as the process of selecting relevant excerpts (sentences, paragraphs, etc.) from the original document and concatenating them into a shorter form. Thus, most of recent works in this research area are based on extraction (Goldstein et al., 1999). Although one may argue that extraction approach makes the text hard to read due to the lack of coherence, it also depends on the objective of summarization. If we need to generate summaries that can be used to indicative what topics are addressed in the original document, and thus can be used to alert the uses as the source content, i.e., the indicative function (Mani et al., 1999), extraction approach is capable of handling this kind of tasks.

There have been many researches on text summarization problem. However, in Thai, we are in the initial stage of developing mechanisms for automatically summarizing documents. It is a challenge to summarize these documents, since they are extremely different from documents written in English. Similar to Chinese or Japanese, for the Thai writing system, there are no boundaries between adjoining words, and also there are no explicit sentences boundaries within the document. Fortunately, there is the use of the paragraph structure in the Thai writing system, which is indicated by indentations and blank lines. Therefore, extracting text spans from Thai documents at the paragraph level is a more practical way.

In this paper, we propose a practical approach to Thai text summarization by extracting the most relevant paragraphs from the original document. Our approach considers both the local and global properties of these paragraphs, which their meaning will become clear later. We also present an efficient ap-

proach for solving Thai word segmentation problem, which can enhance a basic word segmentation algorithm yielding more useful output. We provide experimental evidence that our approach achieves acceptable performance. Furthermore, our approach does not require the external knowledge other than the document itself, and be able to summarize general text documents.

The remainder of this paper is organized as follows. In Section 2, we review some related work and contrast it with our work. Section 3 describes the preprocessing for Thai text, particularly on word segmentation. In Section 4, we present our approach for extracting relevant paragraphs in detail, including how to find clusters of significant words, how to discover relations of paragraphs, and an algorithm for combining these two approaches. Section 5 describes our experiments. Finally, we conclude in Section 6 with some directions of future work.

2 Related Work

A comprehensive survey of text summarization approaches can be found in (Mani, 1999). We briefly review here based on extraction approach. Luhn (1959) proposed a simple but effective approach by using term frequencies and their related positions to weight sentences that are extracted to form a summary. Subsequent works have demonstrated the success of Luhn's approach (Buyukkokten et al., 2001; Lam-Adesina and Jones, 2001; Jaruskulchai et al., 2003). Edmunson (1969) proposed the use of other features such as title words, sentence locations, and bonus words to improve sentence extraction. Goldstein et al. (1999) presented an extraction technique that assigns weighted scores for both statistical and linguistic features in the sentence. Recently, Salton et al. (1999) have developed a model for representing a document by using undirected graphs. The basic idea is to consider vertices as paragraphs and edges as the similarity between two paragraphs. They suggested that the most important paragraphs should be linked to many other paragraphs, which are likely to discuss topic covered in those paragraphs.

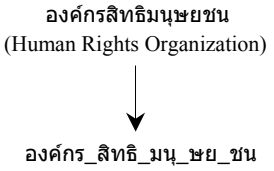
Statistical learning approaches have also been studied in text summarization problem. The first known supervised learning algorithm was proposed by Kupiec et al. (1995). Their approach estimates the probability that a sentence should be included in a summary given its feature values based on the independent assumption of Bayes' Rule. Other supervised learning algorithms have already been investigated. Chuang and Yang (2000) studied several algorithms for extracting sentence segments, such as decision tree, naive Bayes classifier, and neural network. They also used rhetorical relations for representing features. One drawback of the supervised learning algorithms is that they require an annotated corpus to learn accurately. However, they may perform well for summarizing documents in a specific domain.

This paper presents an approach for extracting the most relevant paragraphs from the original document to form a summary. The idea of our approach is to exploit both the local and global properties of paragraphs. The local property can be considered as clusters of significant words within each paragraph, while the global property can be though of as relations of all paragraphs in the document. These two properties can be combined and tuned to produce a single measure reflecting the informativeness of each paragraph. Finally, we can apply this combination measure for ranking and extracting the most relevant paragraphs.

3 Preprocessing for Thai Text

The first step for working with Thai text is to tokenize a given text into meaningful words, since the Thai writing system has no delimiters to indicate word boundaries. Thai words are not delimited by spaces. The spaces are only used to break the idea or draw readers' attention. In order to determine word boundaries, we employed the longest matching algorithm (Sornlertlamvanich, 1993). The longest matching algorithm starts with a text span that could be a phrase or a sentence. The algorithm tries to align word boundaries according to the longest possible matching character compounds in a lexicon. If no match is found in the lexicon, it drops the rightmost character in that text according to the morphological rules and begins the same search. If a word is found, it marks a boundary at the end of the longest word, and then begins the same search starting at the remainder following the match.

In our work, the lexicon contained 32675 words. However, the limitation of this algorithm is that if the target words are compound words or unknown words, it tends to produce incorrect results. For example, a compound word is segmented as the following:



Since this compound word does not appear in the lexicon, it becomes small useless words after the word segmentation process. We further describe an efficient approach to alleviate this problem by using an idea of phrase construction (Ohsawa et al., 1998).

Let w_i be a word that is firstly tokenized by using the longest matching algorithm. We refer to $w_1 w_2 \dots w_n$ as a phrase candidate, if $n > 1$, and no punctuation and stopwords occur between w_1 and w_n. It is well accepted in information retrieval community that words can be broadly classified into content-bearing words and stopwords. In Thai, we found that words that perform as function words can be used in place of stopwords similar to English. We collected 253 most frequently occurred words for making a list of Thai stopwords.

Given a phrase candidate consisting of n words, we can generate a set of phrases in the following form:

$$W = \left\{ \begin{array}{cccc} w_1w_2 & w_1w_2w_3 & \dots & w_1w_2w_3 \dots w_{n-1}w_n \\ & w_2w_3 & \dots & w_2w_3 \dots w_{n-1}w_n \\ & & & \vdots \\ & & & w_{n-1}w_n \end{array} \right\} \quad (1)$$

For example, if a phrase candidate consists of four words, $w_1w_2w_3w_4$, we then obtain $W = \{w_1w_2, w_1w_2w_3, w_1w_2w_3w_4, w_2w_3, w_2w_3w_4, w_3w_4\}$. Let l be the number of set elements that can be computed from $l = (n \cdot (n-1))/2 = (4 \cdot 3)/2 = 6$. Since we use both stopwords and punctuation for bounding the phrase candidate, this approach produces a moderate number of set elements.

Let V be a temporary lexicon. After building all the phrase candidates in the document and generating their sets of phrases, we can construct V by adding phrases that the number of occurrences exceeds some threshold. This idea is to exploit redundancy of phrases occurring in the document. If a generated phrase frequently occurs, this indicates that it may be a meaningful phrase, and should be included in the temporary lexicon using for re-segmenting words.

We denote U to be a main lexicon. After obtaining the temporary lexicon V, we then re-segment words in the document by using $U \cup V$. With using the combination of these two lexicons, we can recover some words from the first segmentation. Although we have to do the word segmentation process twice, the computation time is not prohibitive. Furthermore, we obtain more meaningful words that can be extracted to form keywords of the document.

4 Generating Summaries by Extraction

4.1 Finding Clusters of Significant Words

In this section, we first describe an approach for finding clusters of significant words in each paragraph to calculate the *local clustering score*. Our approach is reminiscent of Luhn's approach (1959) but uses the other term weighting technique instead of the term frequency. Luhn suggested that the frequency of a word occurrence in a document, as well as its relative position determines its significance in that document. More recent works have also employed Luhn's approach as a basis component for extracting relevant sentences (Buyukkokten et al., 2001; Lam-Adesina and Jones, 2001). This approach performs well despite of its simplicity. In our previous work (Jaruskulchai et al., 2003), we also applied this approach for summarizing and browsing Thai documents through PDAs.

Let β be a subset of a continuous sequence of words in a paragraph, $\{w_u \dots w_v\}$. The subset β is called a cluster of significant words if it has these characteristics:

- The first word w_u and the last word w_v in the sequence are significant words.
- Significant words are separated by not more than a predefined number of insignificant words.

For example, we can partition a continuous sequence of words in a paragraph into clusters as shown in Figure 1. The paragraph consists of twelve words. We use the boldface to indicate positions of significant words. Each cluster is enclosed with brackets. In this example, we define that a cluster is created whereby significant words are separated by not more than three insignificant words. Note that many clusters of significant words can be found in the paragraph. The highest score of the clusters found in the paragraph is selected to be the paragraph score. Therefore, the local clustering score for paragraph s_i can be calculated as follows:

$$L_{s_i} = \text{argmax}_{\beta} \frac{ns(\beta, s_i)^2}{n(\beta, s_i)} , \quad (2)$$

where $ns(\beta, s_i)$ is the number of bracketed significant words, and $n(\beta, s_i)$ is the total number of bracketed words.

We can see that the first important step in this process is to mark positions of significant words for identifying the clusters. Our goal is to find topical words, which are indicative of the topics underlying the document. According to Luhn's approach, the term frequencies is used to weight all the words. The other term weighting scheme frequently used is TFIDF (Term Frequency Inverse Document Frequency) (Salton and Buckley, 1988). However, this technique needs a corpus for computing IDF score, causing the genre-dependent problem for generic text summarization task.

In our work, we decide to use TLTF (Term Length Term Frequency) term weighting technique (Banko et al., 1999) for scoring words in the document instead of TFIDF. TLTF multiplies a monotonic function of the term length by a monotonic function of the term frequency. The basic idea of TLTF is based on the assumption that words that are used more frequently tend to be shorter. Such words are not strongly indicative of the topics underlying in the document, such as stopwords. In contrast, words that are used less frequently tend to be longer. One significant benefit of using TLTF term weighting technique for our task is that it does not require any external resources, only using the information within the document.

$$w_1[\boldsymbol{w_2} w_3 \boldsymbol{w_4}]\ w_5 w_6 w_7 w_8[\boldsymbol{w_9} w_{10} \boldsymbol{w_{11}} \boldsymbol{w_{12}}]$$

Figure 1: Clusters of significant words.

4.2 Discovering Relations of Paragraphs

We now move on to describe an approach for discovering relations of paragraphs. Given a document D, we can represent it by an undirected graph $G = (V, E)$, where $V = \{s_1, \ldots, s_m\}$ is the set of paragraphs in that document. An edge (s_i, s_j) is in E, if the cosine similarity between paragraphs s_i and s_j is above a certain threshold, denoted α. A paragraph s_i is considered to be a set of words $\{w_{s_{i,1}}, w_{s_{i,2}}, \ldots, w_{s_{i,t}}\}$. The cosine similarity between two paragraphs can be calculated by the following formula:

$$sim(s_i, s_j) = \frac{\sum_{k=1}^{t} w_{s_{i,k}} w_{s_{j,k}}}{\sqrt{\sum_{k=1}^{t} w_{s_{i,k}}^2 \sum_{k=1}^{t} w_{s_{j,k}}^2}} . \quad (3)$$

The graph G is called the text relationship map of D (Salton et al., 1999). Let d_{s_i} be the degree of node s_i. We then refer to d_{s_i} as the *global connectivity score*. Generating a summary for a given document can be processed by sorting all the nodes with d_{s_i} in decreasing order, and then extracting n top-ranked nodes, where n is the target number of paragraphs in the summary.

This idea is based on Salton et al.'s approach that also performs extraction at the paragraph level. They suggested that since a highly bushy node is linked to a number of other nodes, it has an overlapping vocabulary with several paragraphs, and is likely to discuss topics covered in many other paragraphs. Consequently, such nodes are good candidates for extraction. They then used a global bushy path that is constructed out of n most bushy nodes to form the summary. Their experimental results on encyclopedia articles demonstrates reasonable results.

However, when we directly applied this approach for extracting paragraphs from moderately-sized documents, we found that using only the global connectivity score is inadequate to measure the informativeness of paragraphs in some case. In order to describe this situation, we consider an example of a text relationship map in Figure 2. The map is

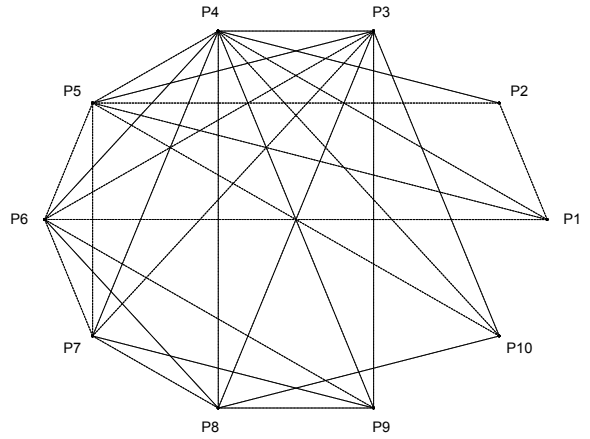


Figure 2: Text relationship map of an online newspaper article using $\alpha = 0.10$.

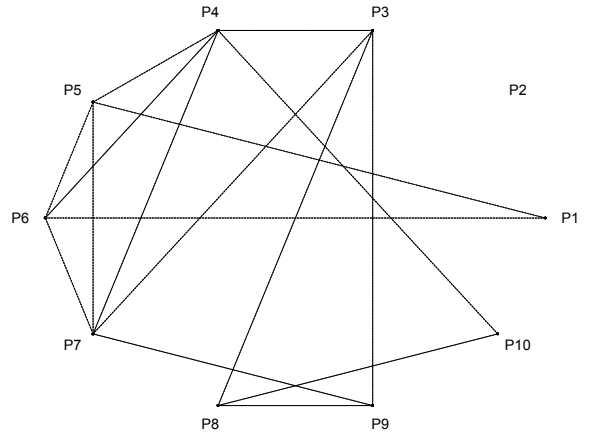


Figure 3: Text relationship map of the same article, but using $\alpha = 0.20$.

constructed from an online newspaper article.[1] The similarity threshold α is 0.1. As a result, edges with similarities less than 0.1 do not appear on the map. Node P4 obtains the maximum global connectivity score at 9. However, the global connectivity score of nodes P3, P5, and P6 is 7, and nodes P7 and P8 is 6, which are slightly different. When we increase the threshold $\alpha = 0.2$, we obtain a text relationship map as shown in Figure 3. Nodes P4 and P7 now achieve the same maximum global connectivity score at 5. Nodes P3, P5, and P6 get the same score at 4.

From above example, it is hard to determine that node P4 is more relevant than nodes such as P3 or P5, since their scores are only different at 1 point. Our preliminary experiments with many other documents lead to the suggestion that the global connectivity score of nodes in the text relation map tends to be slightly different on some document lengths. Given a compression rate (ratio of the summary length to the source length), if we immediately extract these nodes of paragraphs, many paragraphs with the same score are also included in the summary.

4.3 Combining Local and Global Properties

In this section, we present an algorithm that takes advantage of both the local and global properties of paragraphs for generating extractive summaries. From previous sections, we describe two different approaches that can be used to extract relevant paragraphs. However, these extraction schemes are based on different views and concepts. The local clustering score only captures the content of information within paragraphs, while the global connectivity score mainly considers the structural aspect of the document to evaluate the informativeness of paragraphs. This leads to our motivation for unifying good aspects of these two properties. We can consider the local clustering score as the local property of paragraphs, and the global connectivity score as the global property. Here we propose an algorithm that combines the local clustering score with the global connectivity score to get a single measure reflecting the informativeness of each paragraph, which can be tuned according to the relative importance of properties.

Our algorithm proceeds as follows. Given a document, we start by eliminating stopwords and extracting all unique words in the document. These unique words are used to be the document vocabulary. Therefore, we can represent a paragraph s_i as a vector. We then compute similarities between all the paragraph vectors using equation (3), and eliminate edges with similarities less than a threshold in order to build the text relationship map. This process automatically yields the global connectivity scores of the paragraphs. Next, we weight each word in the document vocabulary using TLTF term weighting technique. All the words are sorted by their TLTF scores,

[1]The article is available at: `http://mickey.sci.ku.ac.th/~TextSumm/sample/t1.html`

and top r words are selected to be significant words. We mark positions of significant words in each paragraph to calculate the local clustering score. After obtaining both scores, for each paragraph s_i, we can compute the combination score by using the following ranking function:

$$F(s_i) = \lambda G' + (1 - \lambda)L' \,, \tag{4}$$

where G' is the normalized global connectivity score, and L' is the normalized local clustering score. The normalized global connectivity score G' can be calculated as follows:

$$G' = \frac{d_{s_i}}{d_{max}} \,, \tag{5}$$

where d_{max} is the degree of the node that has the maximum edges using for normalization, resulting the score in the range of $[0, 1]$. Using equation (2), L' is given by:

$$L' = \frac{L_{s_i}}{L_{max}} \,, \tag{6}$$

where L_{max} is the maximum local clustering score using for normalization. Similarly, it results this score in the range of $[0, 1]$. The parameter λ is varied depending on the relative importance of the components G' and L'. Therefore, we can rank all the paragraphs according to their combination scores in decreasing order. We finally extract n top-ranked paragraphs corresponding to the compression rate, and rearrange them in chronological order to form the output summary.

5 Experiments

5.1 Data Sets

The typical approach for testing a summarization system is to create an "ideal" summary, either by professional abstractors or merging summaries provided by multiple human subjects using methods such as majority opinion, union, or intersection (Jing et al., 1998). This approach is known as intrinsic method. Unlike in English, standard data sets in Thai are not yet available for evaluating text summarization system. However, in order to observe characteristics of our algorithm, we collected Thai documents, including agricultural news (D1.AN), general news (D2.GN), and columnist's articles (D3.CA) to make data sets. Each data set consists of 10 documents, and document sizes range from 1 to 4 pages. We asked a student in the Department of Thais, Faculty of Liberal Arts, for manual summarization by selecting the most relevant paragraphs that can indicate the main points of the document. These paragraphs are called *extracts*, and then are used for evaluating our algorithm.

5.2 Performance Evaluations

We evaluate results of summarization by using the standard precision, recall, and F_1. Let J be the number of extracts in the summary, K be the number of selected paragraphs in the summary, and M be the number of extracts in the test document. We then refer to precision of the algorithm as the fraction between the number of extracts in the summary and the number of selected paragraphs in the summary:

$$Precision = \frac{J}{K} \,, \tag{7}$$

recall as the fraction between the number of extracts in the summary and the number of extracts in the test document:

$$Recall = \frac{J}{M} \,. \tag{8}$$

Finally, F_1, a combination of precision and recall, can be calculated as follows:

$$F_1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \,. \tag{9}$$

5.3 Experimental Results

In this section, we provide experimental evidence that our algorithm gives acceptable performance. The compression rate of paragraph extraction to form a summary is 20% and 30%. These rates yield the number of extracts in the summary comparable to the number of actual extracts in a given test document. The threshold α of the cosine similarity is 0.2. The parameter λ for combining the local and global properties is 0.5. For the distance between significant words in a cluster, we set that significant words are separated by not more than three insignificant words.

Table 1 and 2 show a summary of precision, recall, and F_1 for each compression rate, respectively. We can see that average precision values of our algorithm slightly decrease, but average recall values increase when we increase the compression rate.

คำสำคัญ (Keywords):

เพนเทียมเอ็ม, โมบายล์โปรเซสเซอร์, โปรเซสเซอร์, อินเทล, เครื่องโน้ตบุ๊ค, เดสก์ท็อป, เทคโนโลยี, ประสิทธิภาพ, การใช้พลังงาน, บริษัทอินเทล, รุ่นปัจจุบัน

ทำการย่อเอกสารที่ 20% (Summarization result at 20%):

ที่น่าสังเกตก็คือทั้ง "เพรสคอตต์" และ "เพนเทียมเอ็ม" นั้น ได้รับการพัฒนาขึ้นมาบนพื้นฐานสถาปัตยกรรมเดียวกัน นั่นหมายความว่า ตอนนี้เดสก์ท็อปโปรเซสเซอร์และโมบายล์โปรเซสเซอร์ของอินเทลมีประสิทธิภาพและความสามารถทัดเทียมกันแล้ว หรือถ้าจะต่างกันก็คงเล็กน้อย

ในแง่ของเทคโนโลยี อินเทลโมบายล์โปรเซสเซอร์ในปัจจุบันจะเป็นการพัฒนาต่อยอดจากเดสก์ท็อปโปรเซสเซอร์ โดยมีการปรับปรุงให้มีการใช้พลังงานน้อยลง กินไฟน้อยลง ซึ่งเครื่องโน้ตบุ๊คที่ใช้อินเทลโมบายล์โปรเซสเซอร์รุ่นปัจจุบันจะสามารถรันบนแบตเตอรี่ได้นาน 1-4 ชั่วโมง

ในแง่ของการตลาด อินเทลจะทำตลาดโปรเซสเซอร์เพนเทียมเอ็ม ภายใต้ชื่อ "เซนทริโน" (Centrino) โดยวางจำหน่วยเป็นชุดแพ็คเกจ ที่นอกจากจะมีโปรเซสเซอร์แล้ว ยังมีชิปเซ็ตและโมดูลระบบสื่อสารไร้สาย (WiFi) รวมอยู่ด้วย

Figure 4: An example of keywords and extracted summaries in Thai.

Data set	$Precision$	$Recall$	F_1
D1.AN	0.600	0.448	0.509
D2.GN	0.518	0.385	0.431
D3.CA	0.530	0.330	0.404

Table 1: Evaluation results obtained by using compression rate 20%.

Data set	$Precision$	$Recall$	F_1
D1.AN	0.550	0.577	0.555
D2.GN	0.464	0.467	0.453
D3.CA	0.523	0.462	0.488

Table 2: Evaluation results obtained by using compression rate 30%.

Since using higher compression rate tends to select more paragraphs from the document, it increases the chance that the selected paragraphs will be matched with the target extracts. On the other hand, it also selects irrelevant paragraphs to be included in the summary, so precision can decrease. Further experiments on larger text corpora are needed to determine the performance of our summarizer. However, these preliminary results are very encouraging. Figure 4 illustrates an example of keywords and extracted summaries for a Thai document using compression rate 20% . The implementation of our algorithm is now available for user testing at `http://mickey.sci.ku.ac.th/~TextSumm/index.html`. The computation time to summarize moderately-sized documents, such as newspaper articles, is less one second.

6 Conclusions and Future Work

In this paper, we have presented a practical approach to Thai text summarization by extracting the most relevant paragraphs from the original document. Our approach takes advantage of both the local and global properties of paragraphs. The algorithm that combines these two properties for ranking and extracting paragraphs is given. Furthermore, the algorithm does not require the external knowledge other than the document itself, and be able to summarize general text documents.

In future work, we intend to conduct experiments with different document genres. We continue to further develop standard data sets for evaluating Thai text summarization system. Many research questions remain. Since extraction performs at the paragraph level, the paragraph lengths may affect the summarization results. The recent approach for editing extracted text spans (Jing and McKeown, 2000) may also produce improvement for our algorithm. We believe that our algorithm is language-independent, which can summarize documents written in many other languages. We plan to experimentally test our algorithm with available standard data sets in English.

Acknowledgments

This research was supported by the grant of the National Research Council of Thailand, 2002. Many thanks to Tan Sinthurahat (Thammasat University) for manual summarizing the data sets.

References

Banko, M., Mittal, V., Kantrowitz, M., and Goldstein, J. 1999. Generating extraction-based summaries from hand-written summaries by aligning text spans. In *Proceedings of PACLING'99*.

Buyukkokten, O., Garcia-Molina, H., and Paepcke, A. 2001. Seeing the whole in parts: Text summarization for web browsing on handheld devices. *WWW10*.

Chuang, W. T., and Yang, J. 2000. Extracting sentence segments for text summarization: A machine learning approach. In *Proceedings of the 23rd ACM SIGIR*, 152–159.

Edmundson, H. P. 1969. New methods in automatic extraction. *Journal of the ACM*, 16(2):264–285.

Goldstein, J., Kantrowitz, M., Mittal, V., and Carbonell, J. 1999. Summarizing text documents: Sentence selection and evaluation metrics. In *Proceedings of the 22nd ACM SIGIR*, 121–128.

Hahn, U., and Mani, I. 2000. The challenges of automatic summarization. *IEEE Computer*, 33(11):29–35.

Jaruskulchai, C., Khanthong, A., and Tantiprasongchai, W. 2003. A Framework for Delivery of Thai Content through Mobile Devices. *Closing Gaps in the Digital Divide Regional Conference on Digital GMS*. Asian Institute of Technology, 190–194.

Jing, H., Barzilay, R., McKeown, K., and Elhadad, M. 1998. Summarization evaluation methods: Experiments and analysis. *AAAI Intelligent Text Summarization Workshop*, 60–68.

Jing, H., and McKeown, K. 2000. Cut and paste based text summarization. In *Proceedings of the 1st Conference of the North American Chapter of the Association for Computational Linguistics*.

Kupiec, J., Pedersen, J., and Chen, F. 1995. A trainable document summarizer. In *Proceedings of the 18th ACM SIGIR*, 68–73.

Lam-Adesina, M., and Jones, G. J. F. 2001. Applying summarization techniques for term selection in relevance feedback. In *Proceedings of the 24th ACM SIGIR*, 1–9.

Luhn, H. P. 1959. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 159–165.

Mani, I., Firmin, T., House, D., Klein, G., Sundheim, B., Hirschman, L. 1999. The TIPSTER SUMMAC Text Summarization Evaluation. In *Proceedings of EACL'99*.

Mani, I., and Maybury, M. T. 1999. Advances in actomatic text summarization. MIT Press.

Ohsawa, Y., Benson, N. E., and Yachida, M. 1998. KeyGraph: Automatic indexing by cooccurrence graph based on building construction metaphor. In *Proceedings of EAdvanced Digital Library Conference*.

Salton, G., and Buckley, C. 1988. Term weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523.

Salton, G., Singhal, A., Mitra, M., and Buckley, C. 1999. Automatic text structuring and summarization. In Mani, I. and Maybury, M. (Eds.), Advances in automatic text summarization. MIT Press.

Sornlertlamvanich, V. 1993. Word segmentation for Thai in machine translation system. *Machine Translation, National Electronics and Computer Technology Center*, 50–56.