An Automatic Thai Text Summarization Using Topic Sensitive PageRank

Aekkasit Chongsuntornsri[1] and Ohm Sornil[2]
Department of Computer Science
National Institute of Development Administration
Bangkok, Thailand
{[1]aekkasit,[2]osornil}@as.nida.ac.th

Abstract— **The continuing growth of World Wide Web and on-line text collections makes a large volume of information available to users. Automatic text summarization allows users to quickly understand documents. In this paper, we propose an automated technique for single document summary extraction in Thai language which combines content-based and graph-based features and introduce the Topic Sensitive PageRank algorithm as a technique for ranking text segments. A series of experiments are performed using a Thai document collection. The results show the superiority of the proposed technique over reference systems.**

Keywords— **Thai Text Summarization, Topic Sensitive PageRank**

I. INTRODUCTION

The continuing growth of World Wide Web and on-line text collections makes a large volume of information available to users. Text summarization provides users with summaries of document contents, allowing them to quickly understand the main ideas of documents. A number of researchers have proposed techniques for automatic text summarization which can be classified into two categories: extraction and abstraction. The abstraction approach [15] requires fusing and phrasing concepts from original texts which is generally complicated due to the complexity of natural language. The more popular approach is to approximate the summarization task by extracting important segments from the document and presenting them to the reader.

An early work by Luhn [22] returns sentences from a document based on measures of terms' significances and their relative positions in a sentence. Edmundson [9] proposes three additional components which are pragmatic words, title words, and structural indicators, such as sentence locations, for calculating sentence scores. Goldstein et al. [12] combines statistical and linguistic features in sentence score assignment.

A number of research employ supervised learning techniques. Kupiec *et al.* [18] treats text summarization as a statistical classification problem by developing a classification function that estimates the probability of a sentence to be included in the summary. Chuang and Yang [8] study several classification modeling techniques, including decision trees and neural networks, in the context of summarization. Another approach for text segment extraction is the graph-based approach. Salton [27,28] generates a summary from text relation links by constructing a graph from term-based sentence similarity and then traversing nodes in the graph. Mihalcea [23,24] proposes techniques based on PageRank [4] and HITS [17] for ranking sentences in a document.

Our approach comprises two stages of computation. In the first stage, segments are represented by content-based feature vectors. The segment-feature matrix is then compressed into a lower dimensional matrix to uncover hidden association patterns and reduce small variations in segment characteristics by using Singular Value Decomposition (SVD) [19]. In the second stage, segments are represented as nodes, and relationships between two segments whose similarity scores above a threshold are represented as edges in a document graph. A graph search technique is then used to recommend segments to be extracted as summary. The proposed technique naturally combines content-based approach and link-based approach to text segment extraction.

In addition, we introduce the Topic Sensitive PageRank algorithm [6], Topic-Sensitive PageRank is used to calculate multiple rankings for segments of the text using the categorized topics. Its spreading activation process can be applied to identify a set of segments that represent the content of a document.

II. DOCUMENT GRAPH CONSTRUCTION

In order to extract a summary from a document, the document is divided into a set of text segments. A segment is a sequence of words that is delimited by stop marks (".", "?", ";", ":", "-", and whitespace). Each segment is pre-processed by word identification, stopword elimination [11], and stemming [26].

A. Content-Based Features

Features used in the construction of a document graph are collected from various text summarization research. The features can be categorized into 6 groups. Group 1 (features 1 to 6) indicates locations of the segment; Group 2 (features 7 and 8) measures relationships between the segment and the document title; Group 3 (features 9 to 13) are properties of terms in the segment; Group 4 (features 14 and 15) measures relationships between the segment and the entire document; Group 5 (features 16 to 18) takes into account existences of

proper nouns and pronouns; and Group 6 (features 19 to 22) identifies relationships between the segment and significant terms.

1) Segment Id: Segment identifier, normalized by the total number of segments in the document [10,21]

2) Paragraph Id: Paragraph identifier, normalized by the total number of paragraphs in the document [8,10]

3) Paragraph Offset: Segment offset within the paragraph containing it, normalized by the total number of segments in the paragraph [8,16,29]

4) Paragraph Location: Location of the paragraph containing the segment relative to the document (initial, medial, or final) [1,2,18]

5) Segment Location: Location of the segment in the document (1st, 2nd, 3rd, or 4th quarter) [2]

6) Segment Length: Number of terms in the segment, normalized by the length of the longest segment [21,29]

7) Title Word: Number of title words that appear in the segment, normalized by the total number of title words [10,18,21]

8) Sim(segment, title): Cosine similarity between the segment and the title [16,29]

9) Total tf: Sum of term frequencies in the segment [1,16, 21]

10) Average tf: Average term frequency in the segment

11) Total tf × isf: Sum (over all terms in the segment) of each term's frequency multiplied by its inverse segment frequency [16,21]

12) Average tf × isf: Average tf × isf of terms in the segment [21]

13) Total tl × tf: Sum (over all terms in the segment) of each term's total frequency multiplied by their frequencies in the segment [3]

14) Sim(segment, text): Cosine similarity between the segment and the whole document [6]

15) Lexical: Number of terms shared with other segments divided by total number of segments in the document [21]

16) Uppercase: Does the segment contain an uppercase word (yes or no) [1,18]

17) Pronoun: Does the segment contain a pronoun (yes or no) [21,32]

18) Proper Noun: Does the segment contain a proper noun (yes or no) [21]

19) Significance Term: Number of significant terms that appear in the segment, normalized by the total number of significant terms in the document [2,8,10,21]

20) Sim(segment, sig term): Cosine similarity between the segment and the set of significant terms

21) Sig.Term-Luhn: Significant term factor calculated according to Luhn [22]

22) Modified Sig.Term-Luhn: Modified Luhn's significant term factor [5]

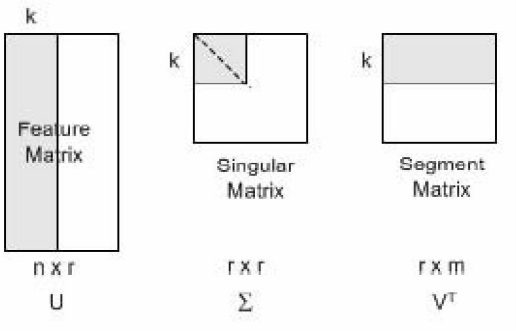

Fig. 1. Singular value decomposition of a segment-feature matrix

B. Segment-Feature Matrix Compression

Each text segment S_j is represented as a vector $<f_{1j}, f_{2j}, \ldots, f_{mj}>$ where f_{ij} is the value of feature i of segment j, m is the number of features, and n is the number of segments in the document. Define A as a segment-feature matrix with n rows and m columns. The organization of the segment-feature matrix assumes that all features are independent which is generally not true in practice. Also, with a large number of features, further processing is computationally expensive due to high dimensionality.

Singular Value Decomposition is used in this research to compress the matrix into a lower dimensional feature space which can uncover hidden relationships among features and segments, and reduce effects of noises in segment characteristics [19].

SVD decomposes matrix A into three components: an orthogonal matrix of singular values, where $r = min(m, n)$, and the left and the right singular vectors (i.e., U and V, respectively), as shown in Figure 1.

By keeping $k < r$ largest values of the singular matrix along with their corresponding columns in U and V, the resulting matrix is the matrix of rank k which is closest to the original matrix A in the least square sense. With respect to this new space of k dimensions, the attributes are no longer independent from each other.

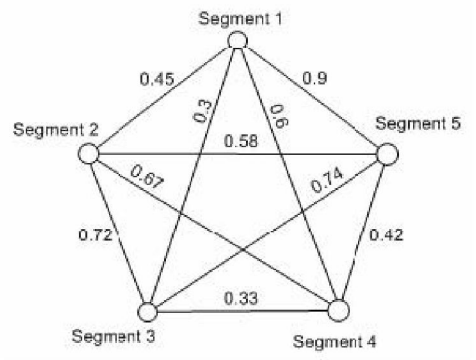

Fig. 2. A document graph

C. Document Graph Construction

The compressed segment-feature matrix is used as the basis for constructing a document graph. Text segments are represented as vertices, and their relationships (whose values are above a threshold) are represented as edges in the graph, as shown in Figure 2.

Formally, let $G = (V, E)$ be a document graph with a set of vertices V and a set of edges (or links) E where $V = \{S_1, S_2, \ldots, S_n\}$; S_i is segment i in the document; and E is a subset of $V \times V$. A segment S_i is defined as a vector $<f_{1i}, f_{2i}, \ldots, f_{ki}>$ where f_{vi} is the value of the content-based feature v of segment i and k is the total number of features in the reduced space. Degree of similarity between segment Si and segment S_j becomes the edge weight between nodes representing the two segments; it can be calculated as follows:

$$similarity \quad (S_i, S_j) = \frac{\sum_{v=1}^{k} f_{S_{iv}} \times f_{S_{jv}}}{\sqrt{\sum_{v=1}^{k} f_{S_{iv}}^2} \sqrt{\sum_{v=1}^{n} f_{S_{jv}}^2}}$$

A graph can be represented as: an undirected weighted graph; a directed weighted graph with orientation of edges set from a sentence to sentences that follow in the text (called forward direction); or a directed weighted graph with the orientation of edges set from a sentence to previous sentences in the text (called backward direction).

III. Text Segment Ranking

Once a graph has been constructed from a document, each segment node will be assigned a significance score, using a graph ranking algorithm. In this research, the Topic Sensitive PageRank algorithm is studied in comparison with Random Walk with Restart and Hopfield Network algorithms which have been used in literature.

A. Topic Sensitive PageRank

Topic Sensitive PageRank (TSPR) [14] is a modification to the original PageRank algorithm [4] which uses multiple PageRank vectors to compute a query-specific rank score for each segment.

The idea of TSPR is to add priority to segments with respect to the topic (in our case, the title of the document). A segment which contains a term in the title is called a special segment. Connections from special segments are assigned higher weights than are those from segments not directly related to the topic.

Each node is initially given a score of $1/|S|$ where $|S|$ is the number of segments in each document. The scoring formula is defined as follows:

$$PR(V_i) = \begin{cases} (1-\alpha)\dfrac{R^{(0)}}{s} + c \times \displaystyle\sum_{V_j \in In(V_i)} \frac{w_{ij} PR(V_i)}{\sum_{V_k \in Out(V_k)} w_{jk}} + E(V_i) & V \in special \\ \cdots & \\ c \times \displaystyle\sum_{V_j \in In(V_i)} \frac{w_{ij} PR(V_i)}{\sum_{V_k \in Out(V_k)} w_{jk}} + E(V_i) & otherwise \end{cases}$$

where s is the number of special segments, $R^{(0)}$ is total number of segments, c is a normalizing constant so that the ranks of all pages always sums to 1, and $E(V_i)$ is a parameter to prevent the "rank sink" problem [7].

B. Hopfield Network Algorithm

Hopfield network algorithm [6] performs a parallel relaxation search, in which nodes are activated in parallel, and activation values from different nodes are combined for each individual node. Neighboring nodes are traversed in order until the activation levels of nodes in the network converge. In the context of a document graph, the graph can be viewed as a network whose nodes are represented by neurons, and edges are represented by synaptic links. The algorithm terminates when there is no significant difference in terms of output between two consecutive iterations.

Initial State: The algorithm is initialized by

$$u_i(0) = 1, \; 0 \le i \le n\text{-}1$$

where $u_i(t)$ is the score of node i at iteration t.

Activation and Update State: Output of each node is calculated as follows:

$$u_i(t+1) = sigmoid[net_j]; \; 0 \le j \le n-1$$

where $net_j = \sum_{i=0}^{n-1} w_{ij} u_i(t)$ is input passed through the activation function, and w_{ij} is the weight of the synaptic link between S_i and S_j, and

$$sigmoid[net_j] = \frac{1}{1 + \exp[\dfrac{\theta_j - net_j}{\theta_0}]}$$

where θ_j is a bias, and θ_0 is an adjustable constant.

Stable State: Repeat the iteration until convergence. The stable state is achieved when sum of the error at every node in the network falls below a given threshold (ε).

$$\sum_{j=0}^{n-1} \left| u_j(t+1) - u_j(t) \right| \le \varepsilon$$

Outputting State: After the network converges, the resulting outputs become the final significance scores of the corresponding segments.

C. Random Walk with Restart

Random Walk with Restart (RWR) [25] is a general method capable of finding correlations between arbitrary modalities of arbitrary text segments. A segment score vector $\vec{u}_q$ can be calculated iteratively by $\vec{u}_q = (1-c)A\vec{u}_q + c\vec{v}_q$ where A is the matrix representation of the document graph with column-normalized; c is the probability of restarting the random walk from the title; $\vec{v}_q$ is a column vector with all its $|S|$ elements zero, except for the entry that corresponds to the

title segment. For each iteration, the estimation of $\vec{u}_q$ is updated until convergence.

IV. Performance Evaluation

In this section, a series of experiments are performed to study the performance of different methods. We randomly select 60 Thai documents, collected from newspaper websites and scientific articles. These documents are manually summarized by 5 readers (common segments are selected as the summary for each document), and 32 English news articles from 7 different natural disasters news sets from the Document Understanding Conference (DUC) collection [9], supplied with reference summaries.

The minimum similarity for establishing an edge in a document graph is 0.2, the value of ε for the Hopfield Network algorithm is 0.00001, the value of c for RWR is 0.66, and the value of $E(V_j)$ for the Topic Sensitive PageRank algorithm is 0.15. The top n% of the segments (also called compression rate) is extracted as summary. The majority of experiments use 20% compression rate, unless stated otherwise.

A. Performance Measures

Precision, recall, and F-measure are used in the evaluation which can be calculated as follows:

$$Recall = \frac{|S_{ref} \cap S_{sys}|}{|S_{sys}|}$$

$$Precision = \frac{|S_{ref} \cap S_{sys}|}{|S_{ref}|}$$

$$F\text{-}measure = \frac{(\alpha+1) * recall * precision}{recall + (\alpha * precision)}$$

where S_{ref} and S_{sys} denote the number of segments appeared in the reference summary and in the system summary, respectively. For F-measure, the experiments use F1 (i.e., the value of α is 1).

In addition, Recall-Oriented Understudy for Gisting Evaluation (ROUGE) [20] is used for the evaluation. ROUGE calculation is based on various statistical metrics by counting overlapping units such as n-grams, word sequences, and word pairs between systems which generate summaries correlating with those extracted by human evaluations. ROUGE-N is an N-gram recall between an automatic summary and a set of manual summaries, which is calculated as:

$$\frac{\sum_{S \in \{Reference Summaries\}} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in \{Reference Summaries\}} \sum_{gram_n \in S} Count_{match}(gram_n)}$$

Among the different values of N, unigram-based ROUGE score (ROUGE-1) has been shown to agree with human judgments the most [21]. In addition, as studied in [30], ROUGE-1 is shown to produce the highest scores for DUC reference systems. Therefore, ROUGE-1 is used in the study, together with precision, recall, and F1 measure.

B. Experimental Results

TSPR is studied against the Hopfield Network and the RWR algorithms on the Thai document collection. Since in Thai language, text is written without spaces between words. A document is thus word-segmented by algorithm presented in [31] prior to the graph construction.

The results, in Tables 1, 2 and 3, show that, according to F1 measure, Topic Sensitive PageRank with backward direction yields the best performance, as also shown in Figure 3.

TABLE I
RESULT OF HOPFIELD NETWORK ALGORITHM

Hopfield Network				
	Recall	Precision	F1	ROUGE(F1)
Undirected	0.21528	0.36974	0.28924	0.18777
Forward	0.17428	0.31385	0.24029	0.08606
Backward	0.36767	0.56160	0.45469	0.32167

TABLE II
RESULT OF RWR ALGORITHM

Random Walk with Restart				
	Recall	Precision	F1	ROUGE(F1)
Undirected	0.29940	0.46960	0.37778	0.26043
Forward	0.22305	0.37228	0.29323	0.07830
Backward	0.23398	0.38771	0.30577	0.13577

TABLE III
RESULT OF TOPIC SENSITIVE PAGERANK ALGORITHM

Topic Sensitive PageRank				
	Recall	Precision	F1	ROUGE(F1)
Undirected	0.29789	0.45433	0.36979	0.31035
Forward	0.17203	0.30526	0.23445	0.12649
Backward	0.39327	0.59625	0.48421	0.36890

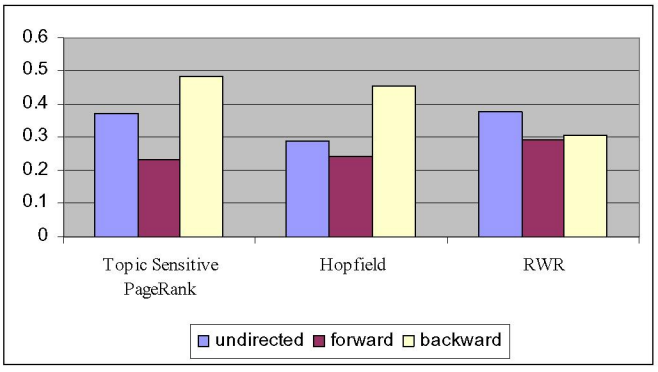

Fig. 3. The F1 results of graph ranking algorithms on Thai dataset

Figure 4 illustrates an example of title and an extracted summary for a Thai document using compression rate of 20%.

Title :
กลุ่มแม่บ้านเกษตรแปลงร่างใยบวบเป็น "รองเท้าขัดผิว"

Summarization result at 20% compression rate

"ใยบวบขัดผิว"ภูมิปัญญาพื้นบ้านซึ่งมีคุณสมบัติช่วยกระตุ้นการสร้างเซลล์ผิวใหม่ กำลังได้รับการพลิกโฉมหน้าครั้งใหญ่เพื่อดึงดูดลูกค้าผู้รักความงามและสุขภาพมากขึ้น เมื่อกลุ่มแม่บ้านเกษตรกรจากลพบุรี แปลงร่างบวบยาวๆ รีๆ เป็นสารพัดผลิตภัณฑ์ขัดผิว โดยเฉพาะ "รองเท้าใยบวบ" ใส่ไปขัดถนอมผิวเท้าไปในตัว

"ผลิตภัณฑ์จากใยบวบ" ที่บรรดาสาวๆ รู้จักประโยชน์กันดีในฐานะตัวช่วยอันสำคัญขณะอาบน้ำล้างผิว ซึ่งกลุ่มแม่บ้านเกษตรกรศิลาทิพย์ จาก ต.ศิลาทิพย์ อ.ชัยบาดาล จ.ลพบุรี ก็ได้นำผลิตภัณฑ์ใยบวบธรรมดาๆ แต่หน้าตาแปลกๆ มาโชว์ในงาน "บิวตี้ แอนด์ เฮลท์ แคร์ แฟร์" (Beauty & Health Care Fair) ณ ศูนย์การแสดงสินค้าเพื่อการส่งออก ไบเทค เมื่อสุดสัปดาห์ที่ผ่านมา

สารพัดผลิตภัณฑ์จากใยบวบที่ทางกลุ่มแม่บ้านนำมาโชว์ในครั้งนี้ เกิดจากการหยิบสิ่งใกล้ตัวใกล้ตาขึ้นมาพลิกโฉมใหม่ให้ดูกิ๊บเก๋ขึ้น จนสามารถคว้ารางวัลสินค้า "หนึ่งตำบล หนึ่งผลิตภัณฑ์" (OTOP-โอทอป) ระดับ 3 ดาวประจำปี 2547 มาครองได้

Fig. 4. An example of the title and the extracted summary

Next, Topic Sensitive PageRank algorithm is studied at 10%, 20%, 30%, and 40% compression rates using ROUGE-1 (F1) measure, as performed in many literatures. The performance scores are shown in Table 4. The results show that higher compression rate gives better performance which corresponds to the intuition.

TABLE IV.
ROUGE-1 SCORES AT DIFFERENT COMPRESSION RATES

Topic sensitive pagerank (F1)				
	10%	20%	30%	40%
Undirect	0.25196	0.31035	0.35127	0.38921
Fwd	0.06848	0.12649	0.17954	0.21616
Bwd	0.15233	0.36890	0.43925	0.46700

Lastly, the algorithms are evaluated on the 32 DUC documents. The results, in Figure 5, also show that TSPR outperforms the other algorithms on this dataset.

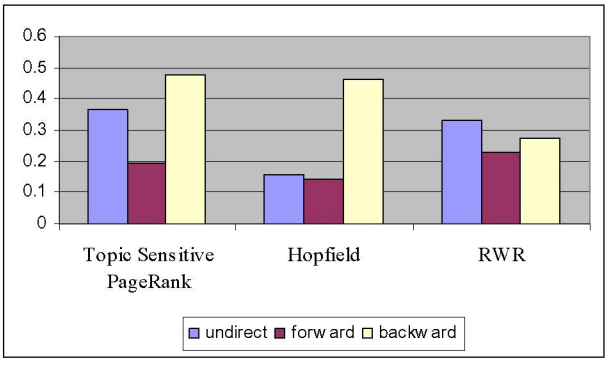

Fig. 5. The F1 results of graph ranking algorithms on the DUC dataset

V. CONCLUSION

This paper presents Topic Sensitive PageRank as a method for computing segment significance scores for automatic Thai text summarization. For each document, a set of content-based features is extracted, reduced, and used in the construction of a document graph whose nodes are then assigned scores representing their significances relative to the content of the document. Experiments are carried out on 60 Thai documents to demonstrate the performance of the proposed method against two other graph ranking algorithms: Random Walk with Restart and Hopfield Network algorithms, used in previous text summarization research. The results show the superiority of the proposed method.

REFERENCES

[1] M. Amini, "The Use of Unlabeled Data to Improve Supervised Learning for Text Summarization," in Proceedings of the 25th SIGIR Conference, Tampere, Finland, 2002.

[2] C. Aone, M. E. Okurowski, and J. Gorlinsky, "Trainable, Scalable Summarization Using Robust NLP and Machine Learning," in Proceedings of the 17th International Conference on Computational Linguistics, Volume 1, Montreal, Quebec, Canada, 1998.

[3] M. Banko, V. Mittal, and M. Kantrowitz, "Generating Extraction-Based Summaries from Hand-Written Summaries by Aligning Text Spans," in Proceedings of PACLING-99, Waterloo, Ontario, 1999.

[4] S. Brin and L. Page, "The Anatomy of A Large-Scale Hypertextual Web Search Engine," Computer Networks and ISDN Systems, 30:1-7, 1998.

[5] Y. Chang, I. Choi, J. Choi, M. Kim, and V. Raghavan, "Conceptual Retrieval Based on Feature Clustering of Documents," 1998.

[6] H. Chen and T. Ng, "An Algorithmic Approach to Concept Exploration in a Large Knowledge Network (Automatic Thesaurus Consultation): Symbolic Branch-and-Bound Search vs. Connectionist Hopfield Net Activation," Journal of the American Society for Information Science, 46(5): 348-369, 1995.

[7] Y-Y. Chen, Q. Gan and T. Suel, "I/O-Efficient Techniques for Computing Pagerank," in Proceedings of the eleventh international conference on Information and knowledge management, Virginia, USA, 2002.

[8] W. T. Chuang and J. Yang, "Extracting Sentence Segments for Text Summarization: A machine Learning Approach," in Proceedings of the 23rd ACM SIGIR Conference, Athens, Greece: 152-159, 2000.

[9] Document Understanding Conference. http//www-nlpir.nist.gov /projects/duc/.

[10] H. P. Edmundson, "New Methods in Automatic Extracting," Journal of the ACM, 16(2): 264-285, 1969.

[11] W. B. Frakes and R. Baeza-Yates, Information Retrieval: Data Structures and Algorithms, Prentice Hall, Englewood Cliffs, NJ, 1992.

[12] J. Goldstein, M. Kantrowitz, V. Mittal, and J. Carbonell, "Summarizing Text Documents: Sentence Selection and Evaluation Metrics," in Proceedings of the 22nd ACM SIGIR Conference, Berkeley, CA: 121128, 1999.

[13] U. Hahn and I. Mani, "The Challenges of Automatic Summarization," IEEE Computer, 33(11): 29-35, 2000.

[14] Haveliwala, T. Topic-Sensitive PageRank. WWW Conference, 2000

[15] Z. Huang, W. Chung, T. Ong, and H. Chen, "A Graph-Based Recommender System for Digital Library," in Proceedings of the 2nd ACM/IEEE-CS Joint Conference on Digital Libraries, Portland, Oregon, USA: 65-73, 2002.

[16] K. Ishikawa, S. Ando, S. Doi, and A. Okumura, "Trainable Automatic Text Summarization Using Segmentation of Sentence," in Proceedings of the Third NTCIR Workshop, Japan, 2003.

[17] J. M. Kleinberg, "Authoritative Sources in A Hyperlinked Environment," Journal of the ACM, 46(5):604-632, 1999.

[18] J. Kupiec, J. Pederson, and F. Chen, "A Trainable Document Summarizer," in Proceedings of the 18th ACM SIGIR: 68-73, 1995.

[19] D. C. Lay, Linear Algebra and Its Applications, 2nd ed. Reading, MA: Addison-Wesley, 1996.

[20] C. Y. Lin and E. H. Hovy, "Automatic Evaluation of Summaries Using N-Gram Co-Occurrence Statistics," in Proceedings of Humane Language Technology Conference, Edmonton, Canada, 2003.

[21] C. Y. Lin and E. H. Hovy, "The Potential and Limitation of Sentence Extraction for Summarization," in Proceedings of the HLT/NAACL Workshop on Automatic Summarization, Edmonton, Canada, 2003.

[22] P. H. Luhn, "Automatic creation of literature abstracts," IBM Journal of Research and Development 2(2). 159-165, 1958.

[23] R. Mihalcea, "Graph-Based Ranking Algorithms for Sentence Extraction, Applied to Text Summarization," in Proceedings of the 42nd Annual Meeting of the Association for Computational Lingusitics, Barcelona, Spain, 2004.

[24] R. Mihalcea and P. Tarau, "TextRank – Bringing Order into Texts," in Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2004), Barcelona, Spain, 2004.

[25] J.-Y. Pan, H.-J. Yang, Faloutsos C., and Duygulu P., GCap:Graph-based automatic image captioning. In Proceedings MDDE '04, 4th International Workshop on Multimedia Data and Document Engineering, Washington, DC, USA, July 2004.

[26] Porter M. F., "An algorithm for suffix stripping," Program, 14(30: 130137, 1980.

[27] G. Salton, C. Buckley, and A. Singhal, "Automatic Analysis, Theme Generation and Summarization of Machinereadable Texts," Science, 264:1421--1426, 1994.

[28] G. Salton, C. Buckley, and A. Singhal, and Mitra M., "Automatic Text Decomposition Using Text Segments and Text Themes," in Proceedings of the 7th ACM conference on Hypertext, Bethesda, Maryland, United States: 53-65, 1996.

[29] D. Shen, Z. Chen, Q. Yang, H-J. Zeng, B. Zhang, Y. Lu, and W-Y. Ha, "Web-Page Classification Through Summarization," in Proceedings of the 27th ACM SIGIR Conference, Sheffield, United Kingdom: 242-249, 2004.

[30] O. Sornil and K. Gree-Ut. "An Automatic Text Summarization Approach using Content-Based and Graph-Based Characteristics." Proceedings of CIS 2006, Bangkok, Thailand, 2006.

[31] O. Sornil, and P. Chiwanarom, "Combining prediction by partial matching and logistics regression for Thai word segmentation" Proceedings of COLING 2004, Geneva, Switzerland, 2004.

[32] P. Turney, "Mining the Web for Lexical Knowledge to Improve Keyphrase Extraction: Learning from Labeled and Unlabeled Data," Journal of Artificial Intelligence Research, 2001