Proceeding of the International Seminar and Conference 2015: The Golden Triangle (Indonesia-India-Tiongkok) Interrelations in Religion, Science, Culture, and Economic.
University of Wahid Hasyim, Semarang, Indonesia. August 28-30, 2015
Paper No. B.8

Indonesian Text Summarization based on Naïve Bayes Method

Ahmad Najibullah
Department of Computer Applied Technology, Nanchang University
999 Xuefu Road, Honggutan New District, Nanchang, Jiangxi Province, P.R. China
Telp:+86-791-83969099 Fax:+86-791-83969069
e-mail: ahmednajibullah@gmail.co

Abstract-*In this paper we present a system for generating summary by sentence extraction. To determine the weight of sentence, we use text features, such as sentence position, sentence relative length, average term frequency, keyword extraction, key phrase extraction, sentence similarity to the title, sentence centrality, inclusion of numerical data, inclusion of entity name, and inclusion of news emphasize words. We also investigate the effect of semantic feature, using latent semantic analysis, on the summarization task. Our experiments show that semantic feature increases precision and F-measure by 9.8% and 2.4% respectively in case of 20% Compression Rate.*

Key words: process, reduce, computer, method

1. Introduction

Automatic Summarization is a process to reduce a document based on computer system in order to generate a summary. Text summarization is the part of the research area in computational intelligence, machine learning and natural language processing. This thesis presents an automatic Indonesian text summarization research. The study is focuses on extraction based summarization approach; sentence extraction and sentence reduction. We use Naïve Bayes approaches to generate a summary. The initial step in summarization is identification of important features. Each document is prepared by pre-processing including sentence segmentation, part of speech tagging, tokenization, stop word removal, and stemming.

Nowadays, information era, Automatic summarization is the way to distillation the important information from a source into a simple form for a particular user or task. Automatic text summarization has been an active research area for many years.

As the problem of overloading online information, the automatic text summarization has made useful applications. A document summary application can become a very valuable tool for humans in understanding the document. Human can save more time to get the important points of document by reading the compressed document, compared to read the full document. Text summarization can be defined as a process of reducing an original text in order to create a condensed form of text that retains the most important points. In order to generate a summary, there are two approaches: abstract and extract, currently most research is relying on extraction to recognizing the most important information in texts.

Evaluation of summarization is a quite hard problem [1]. Even though the date of the automatic text summarization back to Luhn's work in the 1950s, several researchers continue investigating various approaches to the summarization problem up to nowadays [2,3]. Automatic text summarization can be classified into two categories: extraction and abstraction [4]. Extraction summary is a selection of the sentences or phrases from the original text with the highest score and put it together into a new shorter text without changing the source text. The abstraction summary

ISBN 978-602-8273-53-4

www.isc.unwahas.ac.id

method uses linguistic methods to examine and interpret the text. Most of the current automated text summarization system is used to extract method to produce a summary [5]. In summary, the following are the important reasons in support of automatic text summarization:

- A summary or abstract saves reading time
- It improves document indexing efficiency
- Machine generated summary is free from bias
- Customized summaries can be useful in question-answering systems where they provide personalized information.

In this research, our work is a monolingual model, focused in Indonesian language, the official language of Indonesia. With over 230 million speakers, there are so many people talking in Indonesian. Indonesia is the fourth most populous nation in the world. Its large population, the majority speaks Indonesian, making it one of the most widely spoken languages in the world. Therefore, our research is expected to be useful for many people.

2. Related Work

As the problem of data overloading on the internet, we need a useful application to reduce data without loss information. Therefore, summarization text becomes interesting topic. Text Features can be used for as consideration whether the sentence will be retained in a summary [6,7]. Mohamed Abdel Fattah, and Fuji Ren (2008) investigated the effect of the text feature on the summarization task [8]. They used text features score to train a genetic algorithm (GA) and mathematical regression (MR) models to obtain a suitable combination of feature weights. The result of summarization should be grammatical and retain the most important information. Knight Kevin, and Daniel Marcu (2002) proposed noisy-channel and decision-tree approach to solve that problem [9]. Linguistics and statistical methods were used by D.M. Zajic et al. (2008) to create some candidates of summary by multiple sentence compression and then select from them to create the final summary [10]. Specifically, a sentence selector builds the final summary by selecting some candidates, based on features propagated from the sentence compression method, features of the candidates, and features of the summary.

Shiyan Ou et al. (2007) proposed discourse parsing, information extraction and information integration [11]. They used four steps to create summarization; macro-level discourse parsing, information extraction, information integration, and a summary presentation. They selected dissertation abstract in the sociology domain as the source document. The result of that summary was evaluated by comparing the system-generated output against human summary. Masayu Leylia Khodra et al. (2012) proposed automatic tailored multi-paper summarization, combines multi-paper summarization and tailored summarization based on Rhetorical Document Profile, which is a rhetorical structured representation of a paper [12]. Their research is to transform a collection of texts into a single summary by selecting and integrating important contents in the sources.

Kamal Sarkar (2013) used key concepts identified from a document to generate summary by choosing a subset of the sentences from a document that maximizes the important concepts in the final summary [13]. The main point of Kamal's research presented keyphrases in the text for summarization task. He used a simple keyphrase extraction method with two major steps: identifying the candidate keyphrases and ranking the candidate keyphrases for extracting the keyphrases. Wan Xiaojun and Jianguo Xiao (2010) proposed nearest neighborhood knowledge documents to generate document summarization and keyphrase extraction [14]. Their framework consists two steps: neighborhood construction and summary or keyphrase extraction using the neighborhood knowledge.

A machine learning based text summarization has been proposed in Inderjeet Mani et al. (1998) [15], Wesley T. Chuang et al. (2000) [16], Joel Larocca Neto et al. (2002) [17], Fattah (2014), Bijalwan et al. (2014) [18] and some others researcher. Given a training corpus of original documents and their summaries, and the machine learning will be developed the summary by the model obtained.

3. Proposed Method for Indonesian Sentence Compression

3.1 Probabilistic Model

The probabilistic model is used for determining a label to the sentence. Sentence label has two possibilities; *keep and reject*. We can decide the label by comparing both of them; the higher value will be labeled. The naïve Bayes classifier sets a label to a new instance by calculating the probability of each possible value for that label, given the features in the new instance. The summarization is generated by the collection of sentence which the sentence' label is "*keep*". The possibility to a given label from the source sentence might be given as follows:

$$P(label \mid features) \tag{3.1}$$

To decide the label that will be set to the sentence, we have to compare between $P(keep \mid features)$ and $P(reject \mid features)$. To obtain reliable conditional probabilities of the features a huge training set is needed so that every feature is seen many times. Instead, we assume the features are independent. Every single feature has a different characteristic. We can calculate the probability of observing the conjunction of $\{f_1, f_2, ..., f_n\}$ by multiplication of every single feature probability.

$$P(f_1, f_2, ..., f_n \mid keep) = \prod_i (f_1 \mid keep) \tag{3.2}$$

We also use the Bayes rule:

$$P(keep \mid f_1, f_2, ..., f_n) = \frac{P(f_1, f_2, ..., f_n \mid keep) P(keep)}{P(f_1, f_2, ..., f_n)} \tag{3.3}$$

The possibility of $P(f_1, f_2, ..., f_n \mid keep)$ is computed by merging the equation (3.2) and (3.3). By this process, the new equation given as follows:

$$P(keep \mid f_1, f_2, ..., f_n) \propto P(keep) \prod_i P(f_i \mid keep) \tag{3.4}$$

Then the label is decided by the highest value of *argmax* function.

$$P(label \mid f_i) = \underset{label_i \in \{keep, reject\}}{\arg\max} \; P(label_i) \prod_i P(f_i \mid label_i) \tag{3.5}$$

3.2 Naïve Bayes Application

3.2.1 Document Representation

The document D consists of a set of sentences $D = \{S_1, S_2, ..., S_n\}$. Document representation methods can describe the character of the document. The sentence score is calculated by the weighted combination of features. The main objective of this work is to summarize given text. Each sentence is represented by a set of predefined features $(F_1, F_2, ..., F_n)$ then a supervised

learning algorithm is used to train the summarizer to extract important sentence. The feature vector, as shown in table 3.1, is required to perform this process.

Table 3.1 Vector space representation of a document

Sentences	Features				Class
	F_1	F_2	...	F_m	
S_1	x_{11}	x_{12}	...	x_{1m}	y_0 / y_1
S_2	x_{21}	x_{22}	...	x_{2m}	y_0 / y_1
...	...	...	...	...	...
S_n	x_{n1}	x_{n2}	...	x_{nm}	y_0 / y_1

The mathematical model relates output to input as in:

$$[X]*[W]=[Y] \tag{3.6}$$

Where, X is the input matrix (feature parameters). Y is the output vector. W is the linear statistical model of the system (weights $W_1, W_2, ..., W_n$) in:

$$\begin{bmatrix} x_{11} & x_{12} & x_{13} & ... & x_{1m} \\ x_{21} & x_{22} & x_{23} & ... & x_{2m} \\ ... & ... & ... & ... & ... \\ x_{n1} & x_{n2} & x_{n3} & ... & x_{nm} \end{bmatrix} \times \begin{bmatrix} W_1 \\ W_2 \\ ... \\ W_n \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ ... \\ y_n \end{bmatrix} \tag{3.7}$$

X is input matrix; Y is output vector; W is the linear statistical model; x_{ij} is the value of j_{th} feature in i_{th} sentence and W_j is the weight of feature j.

3.2.2 Text Features

Some sentence features such as positive and negative word are extracted by trainable summarization. The trainable summarization is human coding summary. While some others features can be extracted without training process. Feature extraction can be used for representing the important level of sentence [19]. Some of them are thematic features; they are TF-IDf score, keyword extraction, keyphrase extraction, similarity with title, inclusion of numerical, time, and entity data, and centrality. The thematic feature helps the reader more easily understand the document and provide additional information to help the reader comprehend the content [20]. Then location features also used for consideration as part of features extraction. Sentence location and sentence relative length are features that defined by the location in the document.

3.2.2.1 Sentence Position (f_1)

Sentence position is a factor to decide the level of sentence importance in the document. In the Indonesian Language document, the first sentence usually becomes the main idea of the paragraph. The significance of position of sentence plays a vital role in various domains accordingly. In general, every paragraph in a good writing of an article usually only provides one main idea. Thus, we should consider giving a higher value to the first sentence of every paragraph compared to the other sentences. For instance, the first and last sentence in a paragraph has a score value of 5/5, the second sentence has a score 4/5, and so on.

3.2.2.2 Sentence Relative Length (f_2)

The major reason for having difficulty in text comprehension is due to the individuals working memory problem. The reader needs more effort to combine semantics of words present in the sentences. When sentence length is more, it is very hard for a reader to integrate completely and fully understand. Due to short term memory the reader can lose their way in knowing texts. Thus, shorter sentences are penalized [21].

$$f_2 = \frac{length(s)}{\max SentenceLength(d)} \tag{3.8}$$

For example, sentence length is 25, the length of the longest sentence in the document is 55. So, the sentence will get a score value of 0.45.

3.2.2.3 Average TF (f_3)

This feature calculates the Term Frequency (TF) score for each term in a sentence and takes their average. Multiple appearances of a term, except stop-word term, have a good probability to become important content of the document [22].

$$f_3 = \frac{TermFrequency(t,d)}{TermFrequency(d)} \tag{3.9}$$

3.2.2.4 Keyword extraction (f_4)

TF-IDF stands for term frequency-inverse document frequency, often used in information retrieval and text mining. It is a statistical technique used to evaluate how important a word is to a document [23].

$$f_4 = TF * IDF \tag{3.10}$$

where, (3.11)

$$TF_i = \frac{T_i}{\sum_{k=1}^{n} Tk}, \quad IDF = \log \frac{N}{ni}$$

3.2.2.5 Keyphrase extraction (f_5)

Keyphrase extraction is an application used to generate a list of keyphrase automatically. We define the automatic keyphrase extraction as the automatic selection of important, topical phrases from within the body of a document [24]. Sentence inclusion keyphrase has a high probability to include in the summary. This process is performed by matching to the part-of-speech pattern.

$$f_5 = \frac{\#(POS\ patternMatching)}{length(s)} \tag{3.12}$$

3.2.2.6 Sentence Similarity to Title (f_6)

Title contains the group of words that give important clues about text concept. Sentence resemblance to the title is the vocabulary overlap between this sentence and the document title. If the sentence has a higher intersection with the title words, then the score of this feature is higher. It is calculated as follows:

$$f_6 = \left| \frac{\text{keywords in s} \cap \text{keywords in title}}{\text{keywords in s} \cup \text{keywords in title}} \right| \quad (3.13)$$

3.2.2.7 Sentence Centrality (f_7)

This feature considers the term overlap between a sentence and the other sentences in the document. The centrality of the sentence implies its similarity to other sentences. If a sentence has higher centrality, it is assumed to be about an important topic in the document.

$$f_7 = \left| \frac{\text{words in } (s_i) \cap \text{word in other } (s)}{\text{word in the document } (d)} \right| \quad (3.14)$$

3.2.2.8 Inclusion of Numerical Data (f_8)

Numerical data usually carry useful information about the document. Therefore, sentence with the numerical data have a good chance included in the document summary.

$$f_8 = \frac{\#\text{ numerical data } (s)}{length(s)} \quad (3.15)$$

3.2.2.9 Inclusion of Entity Name (f_9)

This feature counts the number of name entities (proper nouns) in a sentence, assuming that a sentence that contains name entities is an important one and have a high probability to include in the summary.

$$f_9 = \frac{\#\text{ entity name } (s)}{length(s)} \quad (3.16)$$

3.2.2.10 Inclusion of News Emphasize Words (f_{10})

This feature counts the number of news emphasize words in a sentence, assuming the sentence has a good chance to be included in the summary. The list of emphasizing words used in the system can be found in the Appendix.

$$f_{10} = \frac{\#\text{ emphasize words } (s)}{length(s)} \quad (3.17)$$

3.2.3 Semantic Feature

The idea of using Latent Semantic Analysis in text summarization is presented by Yihong Gong and Xin Liu [25]. Their method is based on latent semantic indexing, and applied the Singular Value Decomposition (SVD) to generate text summarization.

To illustrate how this method applied in the text summarization, the following process will show the construction of representing single-document in a word-by-sentence matrix. Formally, let A be the $M \times N$ term-document matrix of a collection of documents. $W(|W|) = M$ be the set of keywords in D, and $S(|S| = N)$ be the set of sentences in D. The matrix A can be seen in the equation (3.18), where S_i indicates a sentence and W_i indicates a keyword in the document. In this process, the term that will be processed only term not including in the list of stop words.

$$A = \begin{array}{c|cccc} & S_1 & S_2 & \dots & S_N \\ \hline W_1 & a_{11} & a_{12} & \dots & a_{1N} \\ W_2 & a_{21} & a_{22} & \dots & a_{2N} \\ \dots & \dots & \dots & \dots & \dots \\ W_M & a_{M1} & a_{M2} & & a_{MN} \end{array} \quad (3.18)$$

a_{ij} is the value of TF-IDF score, defined by the equation (3.10). The sentence semantic is determined by using the Singular Value Decomposition (SVD). Then we decompose the matrix A, the SVD of A is defined as:

$$A = UZV^T \quad (3.19)$$

U is an $M \times N$ matrix of left singular vectors, Z is an $N \times N$ diagonal matrix of singular values, and V is an $N \times N$ matrix of right singular vectors. Each column of A is representing a semantic feature of sentence, and each row is representing the semantic word. For the summary generation, we use the semantic sentence representations based on eigenvector of matrix A. The illustration of matrix decomposition can be seen in the figure 3.1. SVD has the capability of mapping m-dimensional term vector space into r-dimensional singular vector space.

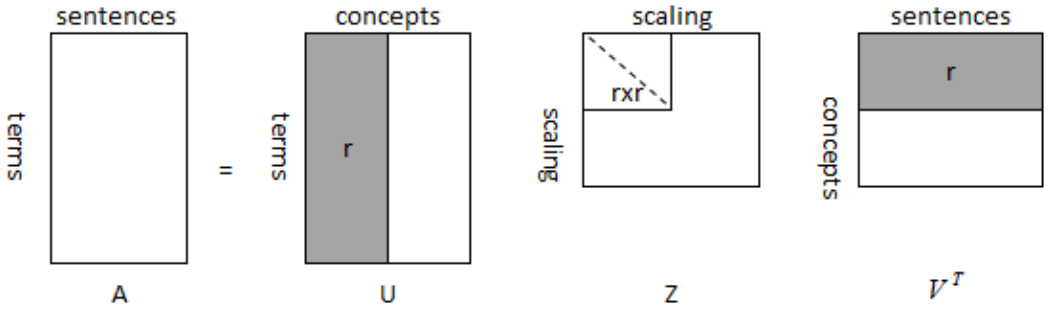

Figure 3.1. Singular Value Decomposition

3.3 The Implementation

3.3.1 Sentence Extraction Process

Below is an example using Naïve Bayes classifier for generating summary. Let F be a new set of features representing a sentence:

$$F = (\text{sent.centrality} = central, \text{sent.location} = pos, \text{keyphrase} = cue)$$

In order to find out whether the above features should be labeled *keep* or *reject* we need to calculate:

$$l_{NB} = \underset{l_i \{keep, reject\}}{\arg\max}\ P(l_i) \prod_l P(f_i \mid l_i) \quad (3.20)$$

with $f_i \in F$ this yields

$$l_{NB} = \underset{l_i \{keep, reject\}}{\arg\max}\ P(l_i) P(sent.cent = central \mid l_j) * P(sent.loc = pos \mid l_j) * P(sent.phrase) = cue \mid l_j) \quad (3.21)$$

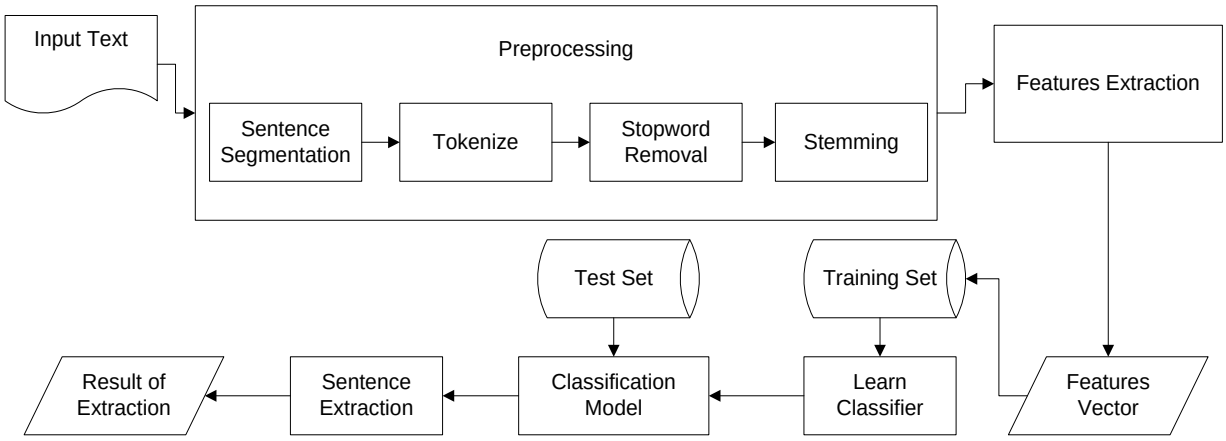

Figure 3.2. Sentence Extraction Process

3.3.2 System Architecture

The sentence score can be calculated by a linear weighted combination of all features. In this work, features are extracted from the text. The summary generation can be seen as a two-class classification problem using the Naïve Bayes classifier, where a sentence is labeled as summary, if it belongs to the extractive reference summary or as not a summary otherwise. In order to perform this extraction, potential features, classifier and a training corpus of the document summary pair are required. The flow diagram of the summary extraction is given in Fig. 3.2.

4. Evaluation Result

The text corpuses used in this project consist of 100 manually summarized documents taken from the Indonesian news website. 80 of these documents were used as the training corpus to train the system. Other 20 documents were used as the testing corpus. The texts included in the corpus were taken from the detik corpus. Detik news is an Indonesian web portal that contains news and online articles, the one of most famous news portals in Indonesia. The statistic of the corpus is shown in Table 5.1.

Table 5.1 Statistic of dataset

Number of docs	100
Average # of sentence per doc	15.83
Maximum # of sentence per doc	28
Minimum # of sentence per doc	6
Summary as (%) of document length	58.64%
Average summary size (in # of sentence)	5.15
Maximum # of sentence per summary	7
Minimum # of sentence per summary	3

Their genre is Newswire. The document consists of 15 sentences in the average. They are manually summarized using a 59% in compression rate. The reference summaries of this text are created manually by an Indonesian native annotator. Figure 5.1 shows an example document and its manual summary generated by human experts.

Extractive summary can be evaluated using various characteristic such as accuracy, cohesion, and readability. Accuracy in extraction measures how far the technique is capable of predicting the correct sentence. Evaluation can be classified into intrinsic and extrinsic evaluation. Intrinsic evaluation judges the summary quality by its coverage between machine-generated summary and human-generated summary. Extrinsic evaluation focuses mainly on the quality by its effect on other tasks. We consider both methods because the generated summary should be informative as well as readable. The former part is objective can be verified using intrinsic evaluation, and the latter part is subjective and can be evaluated using the extrinsic method. In intrinsic evaluation, precision (P), recall (R), and F-measure (F) are used to judge the coverage between the manual and the machine generated summary:

$$P = \frac{|S \cap T|}{|S|} \tag{4.1}$$

$$R = \frac{|S \cap T|}{|T|} \tag{4.2}$$

$$F = \frac{|2 \times P \times R|}{|R + P|} \tag{4.3}$$

where S is the machine generated summary and T is the manual summary. The readability of the summary can be evaluated by existing metrics. Extrinsic evaluation can be done to verify the usability of the summary, by its target audience.

In our experiments, the training and testing are done using 80% and 20% of datasets. In order to investigate the effect of individual features, each feature has to be independent and the correlation between the feature and the class to be good. To analyze the performance of the proposed approach, we investigated the effect of single feature on the summarization task separately. Table 5.3 shows the success rates obtained when individual features are used to summarize the document. The following Table 5.3 shows the effect of individual features in sample dataset. The effect of individual feature's average F-measure is tabulated. The symbols and its explanation are given in Table 5.2.

From the Table 5.3, we can see that the features like numerical data, keyphrase extraction, and sentence relative length have higher score when compared to other features. According to the dataset basically belongs to the Indonesian news document, the significance of numerical data is more in the case of giving information to the reader. The document's keyphrase also give a significant effect to the summary processing. The keyphrase extraction is 77.69% of all features F-measure.

Table 5.2. Notations

No	Symbols	Features
1	SP	Sentence position
2	SRL	Sentence relative length
3	AT	Average TF
4	KwE	Keyword extraction
5	KpE	Keyphrase extraction
6	SST	Sentence similarity to title
7	SC	Sentence centrality
8	IND	Inclusion of numerical data
9	IEN	Inclusion of entity name
10	INE	Inclusion of news emphasize word

Table 5.3 Success rate of individual features

No.	Features	Precision	F-measure	(%) of All Feature
1	SP	0.2442	0.3276	45.81
2	SRL	0.5815	0.5082	71.06
3	AT	0.2559	0.3081	43.08
4	KwE	0.2490	0.3062	42.81
5	KpE	0.5049	0.5556	77.69
6	SST	0.6050	0.3913	54.72
7	SC	0.3373	0.4688	65.55
8	IND	0.7952	0.6582	92.03
9	IEN	0.4821	0.3961	55.39
10	INE	0.4333	0.2456	34.35

We investigated the result of the proposed system. When all feature methods were used, the results of the proposed method were as follows. An evaluation was done by comparing the labels it predicted to the reference summary.

Table 5.4 Performance evaluation of proposed method

	10% of CR	20% of CR	30% of CR
Precision	0.5424	0.6463	0.7385
Recall	0.8500	0.7667	0.7195
F-Measure	0.6483	0.6927	0.7152

From the results given in table 5.4 can be seen that the best result is given in the summary of 30% of the original document. The recall in the CR 10% in the highest one, this condition occurs because the value of false negatives is very high, a large number of sentences are incorrectly labeled *reject* by the naïve Bayes classifier.

Figure 5.2 shows the comparison proposed method to the other methods. Proposed method gives the best F-measure score by 0.7152 in the case 30% of compression rate. This result is better compared to SML method. SML method reaches 0.709 for the best F-measure score.

Figure 5.2 Performance evaluation (F-measure) when compared to other methods.

5. Conclusion

In this work, we have presented the Naïve Bayes method to generate extractive text summarization in Indonesian Language. Naïve Bayes Model is applied to the document based on the text features of the document. The text features that we used in this research are sentence position, sentence relative length, average of term frequency, keyword extraction, keyphrase extraction, sentence similarity to the title, sentence centrality, inclusion of numerical data, inclusion of entity

name, and inclusion of news emphasize word. The result showed that, some features like the inclusion of numerical data, keyphrase extraction, and sentence relative length showed better performance in terms of average summary precision than other features. Moreover, some features like inclusion of news emphasize word and average of term frequency had a lower effect on the performance summary. However the combination of the features gave better results than any of the features. We also executed an experiment for the Latent Semantic Analysis (LSA) effect in the summarization task. The result showed LSA has a good effect to the system performance by increasing 9.8% on the precision score.

We also compared the proposed method to the other researcher method. The proposed method showed the best result compared to the lead, MCBA, GA and SML method. The best F-measure value of the proposed method is 0.7152, while the best score of the other method is 0.709. In the sentence reduction, we compared the proposed method to the Nearest Neighbor method, and the result is the proposed method gave the best result by 18.9%.

In terms of future work, it would be interesting to develop features that are used here, and the model can be tested on different text genres. The corpus, we used in this study consists of news wire documents. However, the test can be run on scientific documents or other genres to see a change in the performance of text features and overall system performance. It would also be worthwhile to try different classifier such as a Nearest Neighbor method to see the difference in the summary result.

6. References

[1] Lloret, Elena. "Text summarization: an overview [J]." *paper supported by the spanish government under the project TEXT-MESS (TIN2006-15265-C06-01)*(2008).

[2] Vishal Gupta, Gurpreet Singh LehalKuceral., A Survey of Text Summarization Extractive, Journal of Emerging Technologies in Web Intelligence [J], vol. 2, no. 3, august 2010.

[3] Hovy, Eduard, and Chin-Yew Lin. "Automated text summarization and the SUMMARIST system [C]." *Proceedings of a workshop on held at Baltimore, Maryland: October 13-15, 1998*. Association for Computational Linguistics, 1998.

[4] Al-Hashemi, Rafeeq. "Text Summarization Extraction System (TSES) Using Extracted Keywords [J]." *Int. Arab J. e-Technol.* 1.4 (2010): 164-168.

[5] El-Haj, Mahmoud, Udo Kruschwitz, and Chris Fox. "Experimenting with automatic text summarisation for Arabic [J]." *Human Language Technology. Challenges for Computer Science and Linguistics*. Springer Berlin Heidelberg, 2011. 490-499.

[6] 李成果. 基于DSC的多文本自动摘要[J]. 计算机系统应用, 2014, (7):7-11. DOI:10.3969/j.issn.1003-3254.2014.07.002.

[7] 邵洲, 张晖. 基于完全稀疏主题模型的多文档自动摘要[J]. 计算机工程与设计, 2014, 35(3):1032-1036. DOI:10.3969/j.issn.1000-7024.2014.03.057.

[8] Fattah, Mohamed Abdel, and Fuji Ren. "Automatic text summarization [J]." *World Academy of Science, Engineering and Technology* 37 (2008): 2008.

[9] Knight Kevin, and Daniel Marcu. "Summarization beyond sentence extraction: A probabilistic approach to sentence compression [J]." *Artificial Intelligence* 139.1 (2002): 91-107

[10] Zajic, David M., Bonnie J. Dorr, and Jimmy Lin. "Single-document and multi-document summarization techniques for email threads using sentence compression [J]." *Information Processing & Management* 44.4 (2008): 1600-1610.

[11] Ou, Shiyan, Christopher Soo-Guan Khoo, and Dion H. Goh. "Design and development of a concept-based multi-document summarization system for research abstracts [J]." *Journal of information science* 34.3 (2008): 308-326.

[12] Khodra, Masayu Leylia, et al. "Automatic Tailored Multi-Paper Summarization based on Rhetorical Document Profile and Summary Specification [J]." *Journal of ICT Research and Applications* 6.3 (2012): 220-239.

[13] Sarkar, Kamal. "Automatic single document text summarization using key concepts in documents [J]." *Journal of information processing systems* 9.4 (2013): 602-620.

[14] Wan, Xiaojun, and Jianguo Xiao. "Exploiting neighborhood knowledge for single document summarization and keyphrase extraction [J]." *ACM Transactions on Information Systems (TOIS)* 28.2 (2010): 8.

[15] Mani, Inderjeet, and Eric Bloedorn. "Machine learning of generic and user-focused summarization." *AAAI/IAAI*. 1998.

[16] Chuang, Wesley T., and Jihoon Yang. "Text summarization by sentence segment extraction using machine learning algorithms [J]." *Knowledge Discovery and Data Mining. Current Issues and New Applications*. Springer Berlin Heidelberg, 2000. 454-457.

[17] Neto, Joel Larocca, Alex A. Freitas, and Celso AA Kaestner. "Automatic text summarization using a machine learning approach [J]." *Advances in Artificial Intelligence*. Springer Berlin Heidelberg, 2002. 205-215.

[18] Bijalwan, Vishwanath, et al. "KNN based Machine Learning Approach for Text and Document Mining [J]." *International Journal of Database Theory and Application* 7.1 (2014): 61-70.

[19] 刘星含，霍华. 基于互信息的文本自动摘要[J]. 合肥工业大学学报：自然科学版，2014，(10):1198-1203. DOI:10.3969/j.issn.1003-5060.2014.10.010.

[20] 胡立. 基于语义层次聚类的多文档自动摘要研究[D]. 华南理工大学, 2014.

[21] Berker, Mine. Using genetic algorithms with lexical chains for automatic text summarization [D]. Diss. Bogaziçi University, 2011.

[22] 覃世安, 李法运. 文本分类中TF-IDF方法的改进研究[J]. 现代图书情报技术, 2013, (10).

[23] 朱平, 费本华, 范少辉等. 基于本体的自动文摘方法研究与实现[J]. 计算机与现代化，2013，(3):34-37. DOI:10.3969/j.issn.1006-2475.2013.03.009.

[24] 马佩勋，高琰. 基于TF* PDF的热点关键短语提取[J]. 计算机应用研究，2013，30(12):3610-3613. DOI:10.3969/j.issn.1001-3695.2013.12.024.

[25] Gong, Yihong, and Xin Liu. "Generic text summarization using relevance measure and latent semantic analysis [C]." *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2001.