Summarizing Text for Indonesian Language by Using Latent Dirichlet Allocation and Genetic Algorithm

Silvia, Pitri Rukmana, Vivi Regina Aprilia, Derwin Suhartono, Rini Wongso, Meiliana
Bina Nusantara University, Computer Science Department
K.H. Syahdan 9, Kemanggisan, West Jakarta 11480, Indonesia
silvia.phang1@gmail.com, pitrirukmana@gmail.com, vieviemoochill@yahoo.com, dsuhartono@binus.edu, rwongso@binus.edu, meiliana@binus.edu

***Abstract* – The number of documents progressively increases especially for the electronic one. This degrades effectivity and efficiency in managing them. Therefore, it is a must to manage the documents. Automatic text summarization is able to solve by producing text document summaries. The goal of the research is to produce a tool to summarize documents in Bahasa: Indonesian Language. It is aimed to satisfy the user's need of relevant and consistent summaries. The algorithm is based on sentence features scoring by using Latent Dirichlet Allocation and Genetic Algorithm for determining sentence feature weights. It is evaluated by calculating summarization speed, precision, recall, F-measure, and some subjective evaluations. Extractive summaries from the original text documents can represent important information from a single document in Bahasa with faster summarization speed compared to manual process. Best F-measure value is 0,556926 (with precision of 0.53448 and recall of 0.58134) and summary ratio of 30%.**

Keywords – Automatic Text Summarization, Sentence Features, Genetic Algorithm, Extractive Summaries, Latent Dirichlet Allocation

I. INTRODUCTION

Information overload has become a problem caused by the easiness of information manipulation, storage, and distribution. Bawden and Robinson (2009) defined the information overload as a term to represent the efficiency of individual state while using the information in their activities becomes hampered due to the massive amount of the available relevant information.

The number of electronic text documents stored in the whole world is uncountable. The Internet development plays a role in the propagation of articles and text documents. Netcraft website survey in August 2013 received responses from 716,822,317 sites. This number has increased of 17,998,808 sites since July 2013. Based on the trend in the last 6 months, Netcraft estimated that there would be 1 (one) billion sites in the next 18 months. Moreover, the number must still be added with the number of electronic text documents that is not available in the Internet. The huge number of text documents available has resulted in demands for a quick access in getting the essence to make decisions based on the available information.

A summary is a text that is produced from one or more texts, that convey important information in the original text(s), and that is no longer than half of the original text(s) and usually significantly less than that (Radev, Hovy, and McKeown, 2002). That is, automatic text summarization is one of the solutions to help finding the core of the document or article in form of a brief description (summary).

According to Jurafsky and Martin (2006), automatic text summarization is the process of distilling the most important information from a text document to create a short version of a task. Research on the implementation of automatic text summarization process continues to develop to this day, especially the extractive summarization. In extractive summarization, there are no changes in the structures of original sentences.

The early research began with the creation of term frequency method by Luhn in 1958 and Baxendale in the same year, followed by the another research done by Edmunson in 1969 (Jezek and Steinberger, 2008). Following those researches, various methods of automatic text summarization turned up, including TF-IDF method (Gupta and Lehal, 2010), Rhetorical Structure Theory (Suneetha, 2011), Cluster-Based (Gholamrezazadeh, Salehi, and Golamzadeh, 2009), Machine Learning (Gupta and Lehal, 2010), Graph (Kumar and Salim, 2012), Latent Semantic Analysis (Gong and Liu, 2001), Sentence Features and the Weighting of Genetic Algorithm (Suanmali, Salim, and Binwahlan, 2011), and Lexical Chains and Genetic Algorithm (Berker and Gungor, 2012).

While for Indonesian text summarization, the various methods are used. They are Graph and Exhaustive Algorithm (Budhi, Intan, Silvia, and Stevanus, 2007), Cluster-Based (SIDoBI by Prasetyo, Uliniansyah, and Riandi in 2008), and Latent Semantic Analysis (Aristoteles, Herdiyeni, Ridha and Adisantoso, 2012). However, the use of Latent Semantic Analysis in the field of Natural Language Processing (NLP) has recently been replaced by Latent Dirichlet Allocation.

The problem found in the existing algorithm for automatic text summarization is about how the algorithm can be used by various parties to create a quick summary from important information more quickly, while it is still maintaining the relevancy and consistency with the original text document.

The creation of the algorithm is expected to help people in creating a summary about important information from text documents more quickly and to satisfy user's need of relevant and consistent summaries through the extraction of important sentences which represent the content of a text document. The algorithm implementation serves as a tool to make a summary of important information from a single Indonesian text document which can be accessed by various parties.

Based on the analysis of previous researches, extractive algorithm will be created to do automatic text summarization for single document in Indonesian language by using sentence features with Latent Dirichlet Allocation and Genetic Algorithm for determining sentence feature weights. This algorithm is based on the sentence features algorithm by Suanmali, Salim, and Binwahlan which was published in 2011 and Genetic Algorithm in the research of Berker and Gungor in 2012 with the addition of Latent Dirichlet Allocation and some modifications.

II. Related Works

Jezek and Steinberger (2008) stated that automatic text summarization began with the publication of sentence extraction from a text using the term frequency method by Luhn in 1958. Method used by Luhn is based on the assumption that word frequency inside a text is an indication of its important level. Some important points which are still currently used are the steps of stemming words into the basic form and followed by the deletion of stop words. In the same year, Baxendale added the idea to use sentence position as one of the determining factors. Baxendale examined 200 paragraphs and found that 85% of the core sentences in the paragraphs are contained in the first sentence, and 7% contained in the last sentence.

The next important development is a method made by Edmunson in 1969 to sum up the weight of term frequency, sentence position, title phrase, and key phrases. The examples of key phrases are "important", "results are", "paper introduces", etc.

Research on automatic text summarization is done continuously and can be divided into several methods. Gupta and Lehal (2010) described the Term Frequency-Inverse Document Frequency (TF-IDF) using the theory that term inside a document is inversely proportional to the number of documents in the corpus that contains that term. One of the extraction systems which use this method is ANES. It was made in 1995.

In the Rhetorical Structure Theory method, Suneetha (2011) explained that the logical connections are different in each parts of the text and interpret the connections. This information referred to the discourse structure and character of the main document.

The next method is Cluster-Based method. Kumar and Salim (2012) defined clustering as grouping similar objects as certain classes. This method is commonly used in multiple document summarization. Other than Cluster-Based Method, there is also machine learning. Gupta dan Lehal (2010) stated that sentences are classified as summary sentence and non-summary sentence based on certain criteria. The classification probability is learnt statistically from training data using Bayes rules, SVM, etc.

Meanwhile for Graph method, Kumar and Salim (2012) stated that graph is used to represent the connection between existing objects. Sentence is an object inside graph and connection is the similarity between the sentences. TextRank is one of the examples of this method.

The other method is the Latent Semantic Analysis. Manning, Raghavan, and Schutze (2009) defined Singular Value Decomposition (SVD) as techniques that can be used to find orthogonal dimension from multidimensional data. SVD is widely used in various fields including the image processing and Latent Semantic Analysis (LSA). One of the examples of its implementation is created by Gong and Liu (2001).

In Genetic Algorithm Based Sentence Extraction for Text Summarization method, Suanmali, Salim, and Binwahlan (2011) extracted the summary by giving score to every sentence features owned by the sentence. Then, they used Genetic Algorithm (GA) in the training process of documents in order to get the weighting or proportion of each features. The method is divided into two stages of preprocessing (the process to cut sentences, tokenization, elimination of stop words, and stemming) and also summarization. In summarization, each sentence will be given value of 0 to 1 for each sentence feature. Sentence features are used as the assessment criteria based on the characteristics of its sentences. These sentence features are title feature, sentence length, term weight, sentence position, sentence to sentence similarity, proper noun, thematic word, and numerical data. Meanwhile, Genetic Algorithm (GA) is used for document training in determining the weight of sentence features.

Another method belongs to Berker and Gungor (2012), i.e. Using Genetic Algorithms with Lexical Chain for Automatic Text Summarization, which used lexical chains and weighting features using Genetic Algorithm. Sentence features used are sentence location (F1), sentence relative length (F2), average TF (F3), average TF-IDF (F4), similarity to title (F5), cue words (F6), named entities (F7), numerical data (F8), sentence centrality (F9), synonym links (F10), and co-occurrence links (F11). The weighting of features is done by using Genetic Algorithm. For each document, features score are calculated per sentence. In the iteration of Genetic Algorithm, initial score of features is defined randomly. The score of sentence is then calculated, and the summary is extracted and evaluated for every document. This process is being repeated and the average precision shows the performance of the iteration. The result of the best iteration will be selected by GA. Each chromosome in the population is the vector of features weight with binary representation. The length of 48 bits chromosome represents 12 features, where each feature has a value between 0 and 15 and represented in 4 bits. Total chromosomes in the population are 1000. For every generation, matching/crossover operator choose 50 chromosomes with the highest fitness and insert them into a new population for the next generation. The rest 950 chromosomes would be produced by parents and are chosen through roulette wheel weighting. This algorithm is

executed for 100 generations and the best chromosome will be chosen as the features weight.

There are several types of summarization task that have been addressed, such as single document summarization, multi-document summarization, summarization focused by question, and headline generation (Nenkova). Researchers conducted aforementioned methods through different type of summarization task to get different result and analyze methods used. For example, Garcia (2009) used n-grams and maximal frequent word sequences as features in a vector space model in order to determine the advantages and disadvantages for extractive text summarization for both single and multi-document summarization tasks. In single-document summarization, the summary of only one document is built, while in multi-document summarization the summary of a whole collection of documents (such as all today's news or all search results for a query) is built. Another research by Mihalcea (2005) that examined a method of language independent extractive summarization that relies on iterative graph-based ranking algorithms for single-document summarization task for English and Portuguese. As a preliminary work, single-document summarization task will be conducted on this research.

The methods mentioned above are the methods used for English. Meanwhile, the research of automatic text summarization for Indonesian has not been much done. Some of them are the Graph method and Exhaustive Algorithm belonging to Budhi, Intan, Silvia, and Stevanus (2007) which used the concept of virtual graph. The process includes the using of TF-IDF and exhaustive algorithm to create a graph. Prasetyo, Uliniansyah, and Riandi (2008) created an application, namely SIDoBI, which is capable to summarize document into an abstract (summary). This application used MEAD which uses cluster centroids method.

Aristoteles, Herdiyeni, Ridha and Adisantoso (2012) created Automatic Text Summarizer for Indonesian using the Genetic Algorithm with 11 considered components, which are sentence position (f1), positive keyword in a sentence (f2), negative keyword in a sentence (f3), similarity with another sentence (f4), similarity with title (f5), the existence of name entity (f6), the existence of numerical data (f7), the relative length of a sentence (f8), path from a node (f9), the summation of the resemblance for each node (f10), and the latent semantic component (f11). The analysis of features weight shows that by using positive keyword in a sentence (f2), similarity with another sentence (f4), similarity with title (f5), and the latent semantic component (f11), are enough to create similar result compared to the result of using all eleven features. All the components are used in training of Genetic Algorithm model to obtain the appropriate weight combination for every component.

III. METHODOLOGY

The algorithm is designed based on the scoring of sentence features in Genetic Algorithm Based Sentence Extraction for Text Summarization by Suanmali, Salim, and Binwahlan (2011), and also the implementation of Generic Algorithm to weight sentence features in Using Genetic Algorithm with Lexical Chains for Automatic Text Summarization which belongs to Berker and Gungor in 2012. There were some modifications like the use of LDA topic modeling, lemmatization methods of Suhartono, Christiandy, and Rolando (2014) method which replace stemming and other adaptations to handle Indonesian language text.

The implementation consists of two stages of training and testing. Training is the stage to determine the weight of sentence features (involving the process to read text input, pre-summarization, summarization, and Genetic Algorithm to generate learned sentence feature weights). Meanwhile, testing stage is the stage to create the summary of text (read text input, pre-summarization, summarization, and saving summary).

In training stage, the first process is to input title, document content, and ratio which will be validated first. The second process is the pre-summarization which includes the separation of text document's content into paragraphs, NLTK tokenizer for sentence and word tokenization, conversion into lower letter case, elimination of stop words, and lemmatization with dictionary lookup into Indonesian Dictionary in MySQL database. The third process is summarization which consists of the calculation of TS-ISF features score, sentence location, and relative length of a sentence, LDA topic modeling, title similarities, keyword similarities, sentence cohesion, and numerical data. The calculation of title similarities, keyword similarities, and sentence cohesion is LDA Topic Modeling and Jensen-Shannon Divergence.

The total score from the features of each sentence will be calculated and some sentences with the highest scores will be extracted according to the ratio of input summaries. Features score of each sentence will be used in training the Genetic Algorithm to find the weight of each feature.

In the process of summarization, there is no difference between training and testing stages, except in the process of sentence extraction. In training stage, each features score of a sentence will be accommodated to be input in determining the feature weights by Genetic Algorithm. Genetic Algorithm defined that the number of generation is 100 with 1000 chromosomes population represented by binary with length of 28 bits (every 4 bits represent weight score of sentence feature with range of 0 to 15). GA also defined fitness function as the average precision of 100 documents while elitist selection will pass 50 chromosomes, the chosen of parents for crossover process through roulette wheel weighting with the crossover weight of 0.8 and mutation rate of 0.2.

In the testing stage, score of sentence is the sum of the features weight multiplication (obtained from training Genetic Algorithm) with each sentence feature score. The weight of sentence features as result from training stage will be used in testing stage. Testing stage provides facility to store summary in form of plain text (.txt) and PDF (.pdf).

The flow of the algorithm framework is described using the flowchart in Fig. 1 (Training Flowchart) and Fig. 2 (Testing Flowchart).

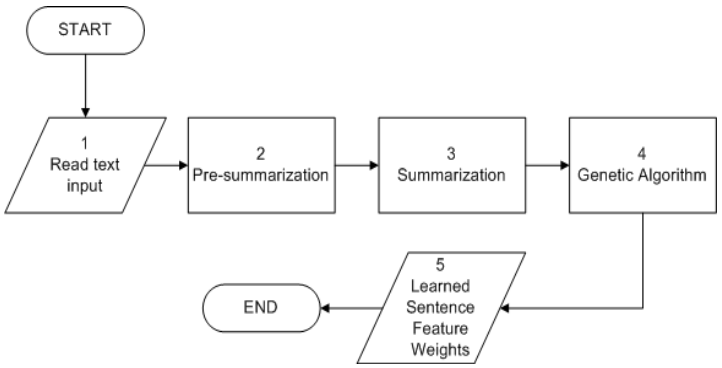

Fig. 1. Training Flowchart of Automatic Text Summarization Algorithm

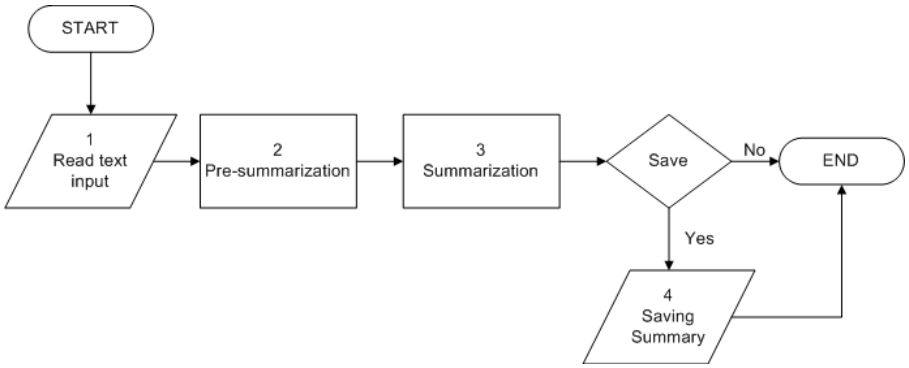

Fig. 2. Testing Flowchart of Automatic Text Summarization Algorithm

In the evaluation step, testing is conducted to examine the speed of summary creation, precision, recall, F-measure, and subjective evaluation.

IV. Results and Discussion

For testing, automatic text summarization for Indonesian Language is implemented using Phyton, web framework Django, package NLTK, and Gensim library in localhost environment. Speed test is done for the creation of 50 documents. The documents are taken from the articles adopted from kompas.com, detik.com, tempo.co, gatra.com, chip.co.id, and femina.co.id. Details of testing are captured in the table I.

TABLE I. Overview of Summary Creation Speed Test

Articles	Fastest Time	Longest Time	Average Time
50 articles	2.395 seconds (5 sentences, 143 words)	3.642 seconds (33 sentences, 571 words)	2.85062 seconds (average of 14.94 sentences, 280.12 words)

It is presented in table 1 above that 50 articles which are tested use 2.85062 seconds as the average processing time. It uses 14.94 sentences and 280.12 words as the average length of articles. It can be concluded that number of sentences and words in a document influence the time needed to create summary, but do not absolutely determine the duration.

Furthermore, testing is done by calculating precision, recall, and F-measure for those 50 articles. The summary references used are manual summary that is done by 29 people using ratio between 15% and 30%. Meanwhile, the summary system is created with three ratios of 10%, 20%, and 30%.

Table II presents the summary of precision, recall, and F-measure test to 50 documents with summary system ratio of 10%, 20%, and 30%. F-measure is calculated by using the formula:

$$F = \frac{2 * P * R}{P + R}$$

TABLE II. Precision, Recall, and F-measure of 50 Articles

Ratio	Average Precision (P)	Average Recall (R)	Average F-measure (F)
10%	0.66666	0.258	0.372025
20%	0.60274	0.4213	0.495946
30%	0.53448	0.58134	0.556926

It can be concluded that the average value of precision is decreasing while the average recall value is improving as the increasing of summary ratio. F-measure is used to define the quality of system summary by combining the precision and recall. It can be seen that the highest F-measure value can be obtained using summary ratio of 30%.

F-measure value of 0.556926 and precision of 0.53448 for a summary with 30% ratio is higher than:

1. F-measure value from sentence features method using Latent Semantic Analysis which is done by Aristoteles, Herdiyeni, Rida, and Adisantoso (2012). It is 0.4763 for the ratio of 30%.
2. F-measure value from sentence features and Genetic Algorithm by Suanmali, Salim and Binwahlan (2011). It is 0.45359 and 0.46471 for the precision.
3. Precision value from Lexical Chains and Genetic Algorithm by Berker and Gungor (2013). It is 0.46.

Subjective evaluation is done by using 55 original articles and summary ratio of 30% from testing stage which are provided to the public through questionnaire. Questionnaires are distributed through Google Docs spreadsheet. Users are asked to assess whether the summary has represented the most important information from original article or not. Choices of answer available in the questionnaires are "not representative at all", "not representative", "enough", "representative", and "very representative".

Total responses received for the 55 articles are 645 answers as presented in table III.

TABLE III. SUBJECTIVE EVALUATION QUESTIONNAIRE RESULT

Answer Choices	Response Number	Percentage (%)
Not representative at all	5	0.775
Not representative	35	5.426
Enough	154	23.876
Representative	330	51.163
Very Representative	121	18.760

From the result above, it can be concluded that 69.923% (total from "representative" and "very representative") are positive response to the representation of the original articles by the system summaries with ratio of 30%. The negative responses are 6.201% (total from "not representative at all" and "not representative") and neutral responses are 23.876%. The significance difference between positive, negative, and neutral responses shows that users felt that the quality of the summaries in major are satisfactory because the summaries have represented the important information needed from the original articles.

V. CONCLUSION AND SUGGESTION

According to the test results and discussion of this automatic text summarization algorithm for Indonesian language, it can be concluded that the algorithm can produce extractive summary which represents important information from a single Indonesian text document more quickly (around 2.395 to 3.642 seconds for text document which consists of 5 to 33 sentences). From the three summary ratios tested, the highest F-measure value can be obtained by the summary with ratio of 30%, with F-measure value of 0.556926, precision of 0.53448 and recall of 0.58134. It is higher than the previous researches.

The suggestions that could be considered for the improvement or development of further research related to the automatic text summarization algorithm for Indonesian language are the additional of feature proper noun for Indonesian and the increasing accuracy of lemmatization algorithm used in pre-summarization process to be more than 98%. This automatic text summarization algorithm can also be used as the basic to develop algorithm for multiple document summarization for Indonesian language. Redundancy analysis plays an important role in that type of summarization. Other than that, corpus document and ideal summary for Indonesian language should also be made by professionals to achieve the standardization of automatic text summarization algorithm testing and evaluation.

ACKNOWLEDGMENT

1. Mr. Marcus Bambang Walgito as Indonesian language lecturer in Bina Nusantara University
2. All respondents that have participated in the evaluation of algorithm implementation.

REFERENCES

[1] Aristoteles, Hardiyeni, Y., Ridha, A., and Adisantoso. "Text Feature Weighting for Summarization of Documents in Bahasa Indonesia Using Genetic Algorithm". International Journal of Computer Science Issues 9(1):1-6, 2012.

[2] Bawden, D. and Robinson, L. "The Dark Side of Information: Overload, Anxiety and Other Pathologies". Journal of Information Science 35(2):180-191, 2009.

[3] Berker, M. and Gungor, T. "Using Genetic Algorithms with Lexical Chains for Automatic Text Summarization". 4th International Conference on Agents and Artificial Intelligence, 1:595-600, 2012.

[4] Budhi, G.S., Intan, R., Silvia, R., and Stevanus, R.R. "Indonesian Automated Text Summarization". Proceeding 1st International Conference on Soft Computing, Intelligent System and Information Technology, 2007.

[5] Garcia, R.A., Hernandez, and Ledeneva, Y. "Word Sequence Models for Single Text Summarization". 2nd International Conferences on Advances in Computer-Human Interactions page 44-48, 2009.

[6] Gholamrezazadeh, S., Salehi, M.A., and Gholamzadeh, B. "A Comprehensive Survey on Text Summarization System". Proceedings of CSA 9:1-6, 2009.

[7] Gong, Y. and Liu, X. 2001. "Generic Text Summarization Using Relevance Measure and Latent Semantic Analysis". Proceedings of The 24th International ACM SIGIR Conference on Research and Development in Information Retrieval 19-25, 2001.

[8] Gupta, V. and Lehal, G.S. "A Survey of Text Summarization Extractive Techniques". Journal of Emerging Technologies in Web Intelligence 2(3):258-268, 2010.

[9] Jezek, K. and Steinberger, J. "Automatic Text Summarization (The State of The Art 2007 and New Challenges)". Znalosti p.1-12, 2008.

[10] Jurafsky, D. and Martin, J.H. "Speech and Language Processing: An Introduction To Natural Language Processing, Computational Linguistics, And Speech Recognition", 2nd Edition. New Jersey: Pearson Prentice Hall, 2006.

[11] Kumar, Y.J. and Salim, N. "Automatic Multi Document Summarization Approaches". Journal of Computer Science 8(1):133-140, 2012.

[12] Manning, C.D., Raghavan, P., and Schutze, H. "Introduction to Information Retrieval". Cambridge: Cambridge University Press, 2009.

[13] Mihalcea, R. And Tarau P. "A Language Independent Algorithm for Single and Multiple Document Summarization". Proceeding of IJCNLP, 2005.

[14] Nenkova, A. "Automatic Text Summarization of Newswire: Lessons Learned from the Document Understanding Conference". Columbia University, 2005.

[15] Netcraft. "Web Server Survey". Retrieved August 18, 2013, from http://news.netcraft.com/archives/2013/08/09/august-2013-web-server-survey.html, 2013.

[16] Radev, D. R., Hovy, E., and McKeown, K. "Introduction to the Special Issue on Summarization". Computational Linguistics., 28(4):399-408, 2002.

[17] Suanmali, L., Salim, N., and Binwahlan, M.S. "Genetic Algorithm Based Sentence Extraction for Text Summarization". International Journal of Innovative Computing 1(1):1-22, 2011.

[18] Suhartono, D., Christiandy, D., and Rolando. "Lemmatization Technique in Bahasa: Indonesian Language". Journal of Software 9 (5), 1202-1209, 2014.

[19] Suneetha, S. "Automatic Text Summarization: The Current State of The Art". International Journal of Science and Advanced Technology 1(9):283-293, 2011.