IJCCS (Indonesian Journal of Computing and Cybernetics Systems)
Vol.13, No.2, April 2019, pp. 137~148
ISSN (print): 1978-1520, ISSN (online): 2460-7258
DOI: 10.22146/ijccs.38261

Automatic Text Summarization Based on Semantic Networks and Corpus Statistics

Winda Yulita*[1], Sigit Priyanta[2], Azhari[3]
[1]Master Program of Computer Science; FMIPA UGM, Yogyakarta, Indonesia
[2,3]Departement of Computer Science and Electronics, FMIPA UGM, Yogyakarta, Indonesia
e-mail: ***[1]winda.yulita@mail.ugm.ac.id**, [2]seagatejogja@ugm.ac.id, [3]arisn@ugm.ac.id

Abstrak

Salah satu metode peringkasan teks otomatis yang sederhana dan dapat meminimalkan redundansi pada ringkasan adalah metode Maximum Marginal Relevance (MMR). Metode MMR memiliki kelemahan yaitu terdapat bagian-bagian yang terpisah satu sama lain dalam hasil ringkasan yang secara semantic tidak terhubung. Oleh karena itu, penelitian ini bertujuan untuk membandingkan hasil ringkasan menggunakan metode MMR berbasis semantic dan MMR berbasis non-semantic. Metode MMR berbasis semantic memanfaatkan WordNet Bahasa dan corpus dalam pemrosesan ringkasan teks. Metode MMR berbasis non-semantic menggunakan metode TF-IDF. Penelitian ini juga melakukan pemampatan ringkasan sebesar 30%, 20% dan 10%. Data penelitian yang digunakan berupa 50 teks berita online. Pengujian hasil ringkasan teks dilakukan dengan menggunakan toolkit ROUGE.. Hasil penelitian menyatakan bahwa nilai rata-rata f-score terbaik pada metode MMR berbasis semantic adalah 0,561, sedangkan nilai f-score terbaik pada metode MMR berbasis non-semantic adalah 0,598. Nilai tersebut dihasilkan dengan menambahkan proses preprocessing berupa stemming dan pemampatan hasil ringkasan 30%. Perbedaan nilai yang diperoleh disebabkan oleh ketidaklengkapan WordNet Bahasa dan terdapat beberapa kata di dalam judul berita yang tidak sesuai dengan EYD (KBBI).

Kata kunci*— peringkasan teks otomatis, metode MMR, semantic, non-semantic*

Abstract

One simple automatic text summarization method that can minimize redundancy, in summary, is the Maximum Marginal Relevance (MMR) method. The MMR method has the disadvantage of having parts that are separated from each other in summary results that are not semantically connected. Therefore, this study aims to compare summary results using the MMR method based on semantic and non-semantic based MMR. Semantic-based MMR methods utilize WordNet Bahasa and corpus in processing text summaries. The MMR method is non-semantic based on the TF-IDF method. This study also carried out summary compression of 30%, 20%, and 10%. The research data used is 50 online news texts. Testing of the summary text results is done using the ROUGE toolkit. The results of the study state that the best value of the f-score in the semantic-based MMR method is 0.561, while the best f-score in the non-semantic MMR method is 0.598. This value is generated by adding a preprocessing process in the form of stemming and compression of a 30% summary result. The difference in value obtained is due to incomplete WordNet Bahasa and there are several words in the news title that are not in accordance with EYD (KBBI).

Keywords*—automatic text summarization, MMR method, semantic, non-semantic*

Received August 23th,2018; Revised April 24th, 2019; Accepted April 25th, 2019

1. INTRODUCTION

The development of the World Wide Web encouraged very rapid information growth, even according to Khan and Salim [1], data on the World Wide Web grew at an exponential pace. Exponential information growth results in information overload on the internet, as is the case with online news texts. Therefore, automatic text summarization is needed to shorten the time in knowing the content of the news.

Automatic text summarization is the process of making a text summary that stores important information and contains general meaning from a text [2]. Its goal is producing text for the given text without loss in the overall information on the source text [3]. Text summarization methods are divided into two categories: extractive and abstractive. Extractive summarization extracts important sentences from source documents and them together to generate the summary. Abstractive summarization creates a brief useful summary by generating new sentences [4]. Some researchers have conducted automatic text summarization in Indonesian with several methods, including summarizing text using sentence scoring and decision trees [5], Text Summarization Based on Semantic Analysis [6], Sentence structure-based summarization [7], dan query-based summarization [8]. The study did not apply word order calculations in sentences. This is important because the same words in sentences but in different sequences sometimes produce different meanings.

Another method that can be applied to Indonesian text is Maximum Marginal Relevance (MMR). The Maximum Marginal Relevance (MMR) method is one of the simplest, most effective and able to reduce redundancies in the text summary results [9]. Other researchers have different opinions regarding text summarization with the MMR method. According to Yapinus, et al. [10], text summarizing using the MMR method alone without applying natural language comprehension techniques will have separate sections in summary results.

2. METHODS

2.1 Research Design

The design of automatic text summarization research that will be built begins with entering the news text. In general, the research design consists of three stages: the text preprocessing stage, similarity measure, and text summarization with the MMR method. Preprocessing is a stage to produce a set of words that are ready to be processed and used as input at a later stage. The similarity measure is a stage to calculate the similarity between sentences. At this stage, the measurement is divided into two parts, namely semantic and non-semantic based measurements. Semantic-based measurement is the calculation of similarities between sentences that will involve the lexical database and corpus. Non-semantic-based measurements only calculate the word distribution in the text with the TF-IDF method and calculate the similarity between sentences using the Cosine Similarity method. The purpose of grouping the similarity measure method is to compare the summary f-score values generated by the text summarization system.

Testing the results of the summary text using the ROUGE (Recall-Oriented Understanding for Gisting Evaluation) toolkit. The ROUGE Toolkit is an N-gram based method that has proven to be highly correlated with human evaluation [11]. This toolkit works by comparing summaries generated by the system with manual summaries.

2.2 Preprocessing

The preprocessing stage is carried out by processing raw documents into documents that are ready to be processed for the next step. The first preprocessing in this study is sentence segmentation which is the breakdown of paragraphs into sentences. The solution is done by separating the sentence based on punctuation (.), Question mark (?) And exclamation point (!). The second preprocessing process is case folding which converts all letters into lowercase

letters and removes characters other than letter characters. The third process is tokenizing which is the process of cutting sentences into words based on spaces. The fourth preprocessing process is filtering which is the process of removing words that are not too influential (stopword) in the text. Furthermore, the stemming process aims to turn the word into a basic word.

2.3 Similarity Measure

The similarity measure is a functional tool used to measure similarities between objects. The result of the similarity measure process is a numerical value between 0 to 1. A value of 0 means very different, while a value of 1 means exactly the same. Similarity measures are grouped into two, namely semantic and non-semantic based measurements. Semantic-based measurements can be applied to find the similarity between sentences and the similarity between words, such as research conducted by Li, et al. [12]. Li, et al. [12] researches the measurement of similarities between sentences by considering semantic information and synthetic information obtained from sentences and words.

2.3.1 Semantic Based Measurement

Semantic-based measurement is a measurement of similarity between sentences involving lexical databases and corpus. The Lexical database used is WordNet Bahasa which is the result of research by Noor et al. [13]. Figure 1 shows a flowchart of a semantic-based measurement process.

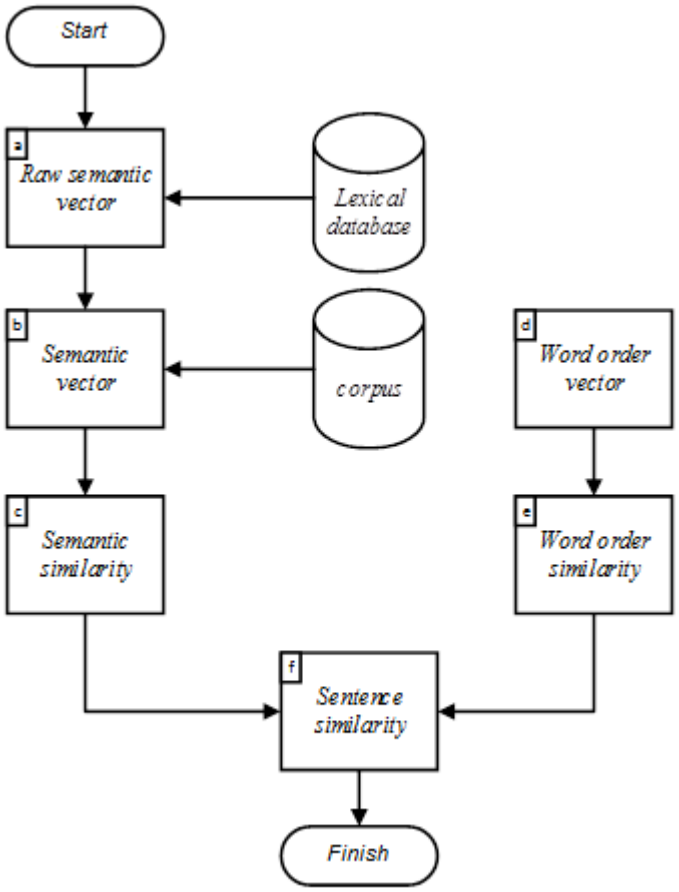

Figure 1 Semantic-based measurement process flow diagram

a. *Raw semantic vector*

The process of raw semantic vector performs calculations about the semantic similarity between words (word similarity). The process of calculating the semantic similarity between words utilizes a lexical database in the form of WordNet Bahasa [13]. WordNet Bahasa is a development of WordNet, so WordNet Bahasa has a hierarchical structure of words that resembles human knowledge and consists of synsets. Synsets is a word or set of words that have exactly the same meaning. This hierarchical structure can determine the semantic distance between words. In Figure 2 there is an example of a semantic hierarchy between words in WordNet.

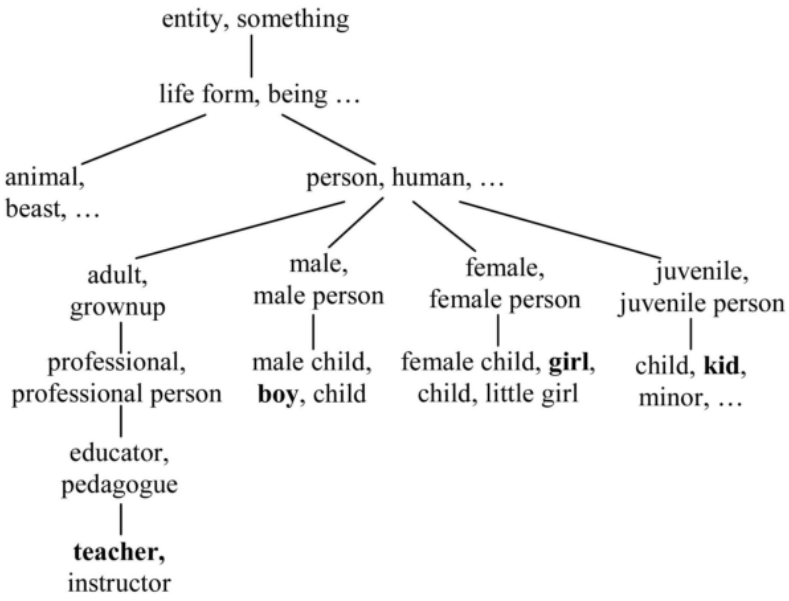

Figure 2 The hierarchical basis of semantic knowledge [12]

So, the semantic similarity calculation between two words (equation 1) can be done using the path length and path depth obtained from WordNet Bahasa.

$$s(w1,w2) = e^{-\propto l}.\frac{e^{\beta h} - e^{-\beta h}}{e^{\beta h} + e^{-\beta h}} \quad (1)$$

Information:
$s(w1,w2)$ is a semantic similarity between words. l is the length of the path that is short between words (w_1 and w_2), whereas h is the subsumer path depth in semantic nets in a hierarchical manner. $\propto$ and β are constants ($\propto$ = 0,2 dan β = 0,45). Optimal values $\propto$ and β are the results of Li, et al.'s research [12] that uses the knowledge base in the form of WordNet. The process of determining the path length can be done based on the following cases:

a. $w1$ and $w2$ are in the same synset. This statement means that both words have the same meaning and have a path length that is 0.
b. $w1$ and $w2$ are not in the same synset, but the syntax for $w1$ and $w2$ contains one or more common words. Examples, synset boy and synset girl have one common word, child. The purpose of this statement is that some $w1$ and $w2$ have the same features so that the path length between the two words is worth 1.

c. $w1$ and $w2$ are not in the same synset and there are no general words in the synset. In this case, the path length must be calculated. Examples of calculations based on Figure 1 are as follows: the path length between the word teacher and boy is 6. This value is derived from the shortest path length that is passed like teacher-educator-professional-adult-person-male-boy.

Path depth (h)needs to be taken into account because the words in the upper layer have more general concepts and semantic similarity is smaller than the words in the lower layer. Path depth value (h)is obtained by calculating the distance between subsumer with the highest or peak synset. Subsumer is the closest distance that connects between two words. The application can be seen in Figure 1. The words girl and boy are connected by synset person, then the person is subsumer. So, the path depth between the words girl and boy is 2. The path passed is the entity-life form-person.

b. *Semantic vector*

The next process is the semantic vectfior that utilizes a corpus for content information. So, the value of the contribution of a word in the text will be calculated by comparing the word to the corpus. Corpus in this study comes from a collection of texts which are the results of research from Dinakaramani et al. [14]. This process also utilizes lexical semantic vector ($\check{s}$)values as shown in Table 1. In Table 1, the first row is a collection of words from the first sentence T_1, while the first column is a collection of unique words w_i from two sentences. This process will produce semantic vector values for each sentence.

Table 1 Example of a semantic vector calculation process

	khawatir	**bantu**	**presiden**	**soeharto**	**ulang**	$\check{s}$	**Weight**
khawatir	1					1	I(khawatir)/I(khawatir)
bantu		1				1	I(bantu)/Ibantu)
presiden			1			1	I(presiden)/ I(presiden)
soeharto				1		1	I(soeharto)/ I(soeharto)
ulang					1	1	I(ulang)/ I(ulang)
jakarta	0	0	0,044	0,036	0	0	I(jakarta)/ I(jakarta)
kompas	0	0	0,079	0,065	0	0	I(kompas)/I(kompas)
dana	0	0	0,157	0	0	0	I(dana)/I(dana)
banpres	0	0	0	0	0	0	I(banpres)/I(banpres)
instruksi	0	0	0,298	0	0	0,298	I(instruksi)/I(presiden)
inpres	0	0	0	0	0	0	I(inpres)/I(inpres)
kerap	0	0	0	0	0	0	I(kerap)/I(kerap)
penting	0	0	0,241	0,200	0,283	0,283	I(penting)/I(ulang)
politik	0	0	0,365	0	0	0,365	I(politik)/I(presiden)
guna	0	0	0	0	0	0	I(guna)/I(guna)
tahu	0	0	0,057	0,047	0	0	I(tahu)/I(tahu)
rakyat	0	0	0,445	0,365	0	0,445	I(rakyat)/I(presiden)

Determining lexical semantic vector ($\check{s}$)values can be seen from the following cases:

a. If x appears in the first sentence (T_1), $\check{s}_i$ is given a value of 1.

b. If w_i does not exist in T_1, then the semantic similarity value between w_i and all the words in T_1 use equation 1. So, the most similar word between w_i and the word on T_1 will have the highest value (ς). If the resulting value exceeds the threshold (0,2), then $\check{s}_i$= ς. If it is less than the threshold, then$\check{s}_i = 0$.

The last column (Weight) is the most significant information content for weighting by involving the corpus. Next, the process of finding semantic vectors for T_1 and T_2. So, equation 2 is a calculation of semantic vector values involving lexical semantic vector ($\check{s}$) and *information content values*.

$$s_i = \check{s}_i \,.I(w_i)\,.\; I(\widetilde{w}_i) \tag{2}$$

Information:
s_i is a semantic vector. $\check{s}_i$ is a lexical semantic vector value (i = 1,2,3,…). w_i is the word in the joint word set (T), $\widetilde{w}_i$ is the word in the sentence. The use of $I(w_i)$ and allows two words to contribute to similarity based on their respective information content. The decrease in the formula for $I(w_i)$ or $I(\widetilde{w}_i)$ can be seen in equation 3.

$$I(w_i) = 1 - \frac{\log(n+1)}{\log(N+1)} \tag{3}$$

Information:
$I(w_i)$: *information content* word w_i in corpus
n: the number of words w_i in the corpus
N: the total number of words in the corpus

c. *Semantic similarity*

After the calculation of s_i for the two sentences is complete, then the next process calculates the semantic similarity value between the two vectors. Equations are seen in equation 4.

$$S_s = \frac{s_1 \,.\, s_2}{||s_1||\,.\;||s_2||} \tag{4}$$

Information:
S_s: *semantic similarity between two sentences*
s_1 and s_2: semantic vector value of sentence 1 and sentence 2

d. *Word order similarity* between sentences

According to Li, et al. [12], the similarity of word order (word order similarity) in the sentence must be taken into account. This is because the same words with different sentence structures sometimes produce different meanings. This statement can be shown in the following example:
T_1: Adik mengajari kakak di taman
T_2: Kakak mengajari adik di taman

The two sentences consist of the same word but a slightly different wording. If you use the "bag of words" method that is calculating the similarity of sentences based on the same word spread, then those two sentences will be identified to have the same meaning. In terms of, if humans interpret the two sentences, they must have different meanings. Therefore, the wording in the sentence is very important to consider to measure the similarity between sentences.

The process of word order similarity calculation between sentences is done by giving an index number to each word in T_1 and T_2. Index numbers are given according to the order of words in the sentence. The next process is formed word arrangement vectors (r_1 and r_2) for T_1 and T_2 based on a collection of unique words (T). Here's the case in determining the word arrangement vector applied to T_1:

Case 1: If the same word is in T_1, then the entry in r_1 will be filled in with the index number that corresponds to T_1. If nothing is the same, the word that is most similar to $\widetilde{w}_i$ in T_1 will be determined.
Case 2: If the similarity value between w_i and $\widetilde{w}_i$ is greater than the threshold, then the entry of w_i in r_1 will be filled with the index $\widetilde{w}_i$ in T_1.
Case 3: If the search value of both cases fails, then the entry w_i in r_1 is 0.
So, the equation for word order similarity between sentences is:

$$S_r = 1 - \frac{||r_1 - r_2||}{||r_1 + r_2||} \quad (5)$$

Information:
S_r: *word order similarity between two sentences*
r_1 and r_2: word arrangement vector for T_1 and T_2

e. *Overall sentence similarity*

According to Li, et al. [12], the calculation of similarities between sentences contains two components that are considered, namely semantic similarity and word order similarity. Semantic similarity shows lexical similarity, while word order similarity is synthetic information that looks at the order of words in a sentence. So, overall sentence similarity can be calculated by equation 6.

$$Sim(T_1, T_2) = \delta S_s + (1 - \delta) S_r \quad (6)$$

Information:
δ determines the relative contribution of semantic information and word order similarity information for calculating overall sentence similarity. δ is worth $0{,}85$ [12].

2.3.2 Non-Semantic Based Measurement

Similarity measurement at this stage does not utilize lexical databases or corpus. The measurement between sentences is done by calculating the word distribution in the news text. At this stage, the word weighting will use the Term Frequency-Inverse Document Frequency (TF-IDF) method and measure the similarity between sentences using the Cosine Similarity method as equation 4. TF-IDF is one method of weighting words in sentences. TF is the calculation of the occurrence of words in the entire document. IDF is a measure of the importance of a word based on the rarity of its occurrence [15]. After the value of both is obtained, the value will be calculated as multiplication as equation 7.

$$w_{i,j} = tf_{i,j} \; x \; log\left(\frac{N}{df_i}\right) \quad (7)$$

Information:
$w_{i,j}$: the weight of a word i
$tf_{i,j}$: the number of words i in j
df_i: the number of documents containing the word i
N: the total number of documents

2.4 Text Summarization with the MMR Method

Maximum Marginal Relevance (MMR) is one of the many text extraction methods that can be applied to summarize a single-document or multi-document by repeatedly ranking and comparing the similarity between documents. This method was first proposed by Carbonell and Goldstein in 1998. The Maximum Marginal Relevance (MMR) method is one of the simplest,

most effective and able to reduce redundancies in the text summary results [9]. If the text summarization results in a high similarity between sentences, then there is a possibility of redundancy, so a method that can reduce redundancy is needed. Equation 8 is a way of calculating MMR to reduce redundancy.

$$MMR(T_i) = \lambda \times Sim\ (T_i, Query) - (1 - \lambda) \times Sim(T_i, Sum) \quad (8)$$

At each step, the MMR method manages the text Sum which contains the sentence that has been extracted into the summary and manages the collection of sentence T_i that has not been extracted. Each extraction, the sentence with the highest value is added to the summary. Each sentence is given the value of weighting its resemblance to the query which is the title of the text. Sim is a cosine similarity between two feature vectors. λ aims to adjust the value given to emphasize relevance and avoid redundancy. λ is worth between 0 and 1.

The main strength of the MMR method is the ability to produce sentences with new information that can be obtained from equation 8. $Sim\ (T_i, Query)$ aims to measure the similarity of a sentence with a given query. The most relevant sentence will be a summary. Furthermore, the sentence chosen as the next summary is a sentence that is still similar to the query but introduces new information so that it can reduce redundancy in the summary. These objectives can be obtained by calculating the similarity of a sentence with a summary that has been prepared previously ($Sim(T_i, Sum)$).

3. RESULTS AND DISCUSSION

3.1 Similarity Measure (Semantic and Non-Semantic Based Measurements)

Based on the results of the study, the value of the f-score comparison summary results between semantic and non-semantic-based measurements on 30% compression can be seen as Figure 3. Based on Figure 3 it can be concluded that the highest f-score value on non-semantic-based measurements is 0.598, while the highest f-score on semantic-based measurements is 0.561. This is because there are two deficiencies found, namely the deficiencies in the WordNet Bahasa and the data used. Weaknesses in WordNet Bahasa are some words in the title and news text not contained in WordNet Bahasa. This will affect the calculation of the similarity between words.

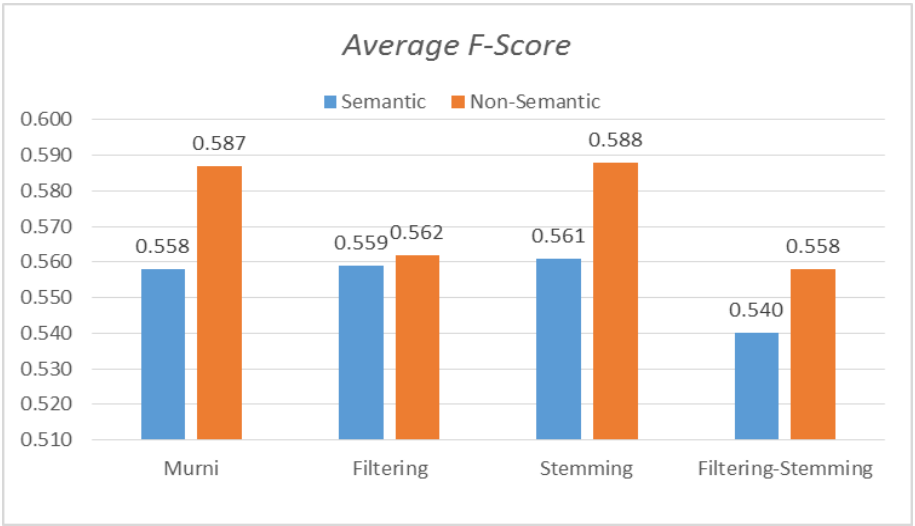

Figure 3 Average semantic and non-semantic based f-score measurements with 30% compression

Furthermore, the lack of data used is the word contained in the news text title does not use standard language, meaning that the word is not in accordance with KBBI (Big Indonesian Dictionary). If a word is not standard (not in accordance with KBBI), then the word will never be in WordNet Bahasa. If a word is not found in WordNet Bahasa, then the value of the similarity between words will produce a zero value. This will affect the results of the summary produced by the text summarization system. This statement can be proven by the example of processing a news text entitled "Akil Mochtar **Persoalkan** Kasasi Praperadilan Ginandjar".

In the news, title there is a word that is not standard (not contained in KBBI) as the word "Persoalkan". The standard word for "Persoalkan" is "Mempersoalkan". The following summary results are carried out by the text summarization system using different words, namely the word "Persoalkan" and "Mempersoalkan" in the news text title:

Table 2 Summarization results with different words

	Manual Summary	**System Summary**	
		"Persoalkan"	**"Mempersoalkan"**
Sentence	1, 2, 20, 21, 22	**1**, 8, 13, **21**, 10	**1**, 8, **20**, **2**,13

In Table 2 it can be seen that the summary generated by the system with different words in the text title ("Persoalkan" dan "Mempersoalkan") produces a different summary. The summary of the system with the title containing the word "Persoalkan" has a summary with the same 2 sentences with the manual summary. These two sentences are the 1st sentence and the 21st sentence. The summary of the system with the title containing the word "Mempersoalkan" has the same 3 sentences with the manual summary. The sentence is the 1st sentence, the 2nd sentence, and the 20th sentence. Based on the summary results produced by the two words, the values of recall, precision and f-score will also be different. Recall, precision, and f-score results look like Table 3.

Table 3 Results of recall, precision, and f-score

Word in Title	***Recall***	***Precision***	***F-Score***
"Persoalkan"	0,611	0,539	0,573
"Mempersoalkan"	0,759	0,741	0,750

Table 3 shows the difference in recall, precision, and f-score values. Recall, precision and f-scores with the word "Mempersoalkan" in the title have a value greater than the recall, precision, and f-score with the word "Persoalkan" in the title. This is because the word "Persoalkan" is not a standard word (according to the KBBI), so the word is not contained in WordNet Bahasa. The influence of a word not found in WordNet Bahasa will result in zero similarity values between words so that it will affect the results of the summary carried out by the text summarization system.

3.2 The Optimal Value in the MMR Method

In this study also look for the optimal value in equation 8. Experiments carried out on Indonesian online news data and calculate the average value of MMR f-score generated as shown in Figure 4.

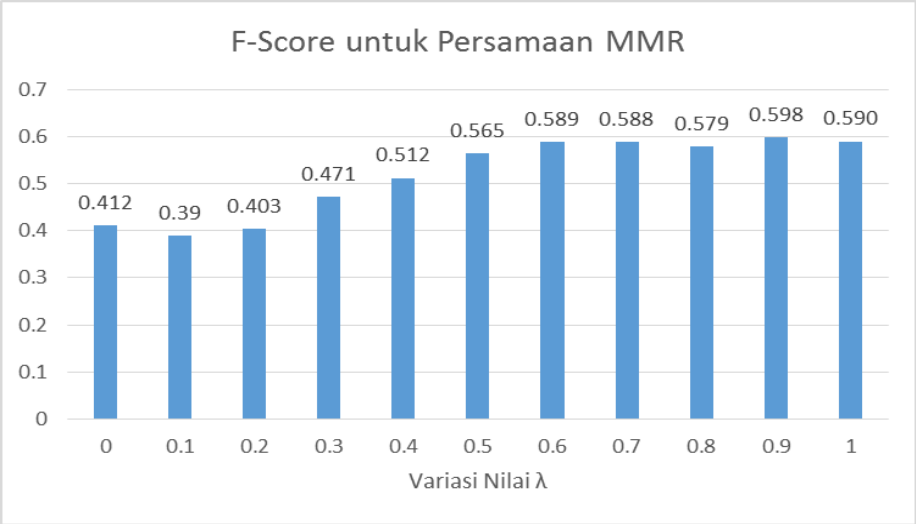

Figure 4 Average MMR based non-semantic f-score with variations in λ values

When λ = 1, MMR gradually calculates the list of standard relevant ratings and calculates the maximum diversity rating between documents at λ = 0. Therefore a linear combination is needed. In Figure 4, the optimal λ value is 0,9. This shows that the summary produced is relevant to the original document.

3.3 Summary Results Based on Summary Compression

In this study, analysis based on summary compression is carried out, namely 30%, 20% and 10% of the length of the text. The results of the study can be seen in Figure 5 and Figure 6. In Figure 5 and Figure 6, it can be seen that the summary results on semantic and non-semantic based measurements and 30% compression of text length resulted in a better f-score than compression of 20% and 10%. from the length of the text. This causes 30% compression to have a longer summary length than 20% and 10% compression, so the chance of similarity between manual summary and system summary is greater.

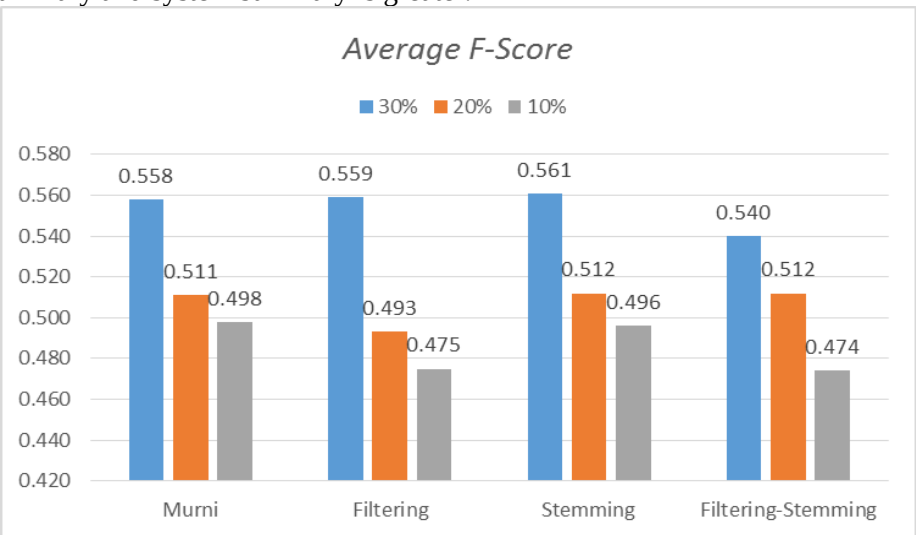

Figure 5 Average semantic-based measurement f-scores with compression of 30%, 20%, and 10%

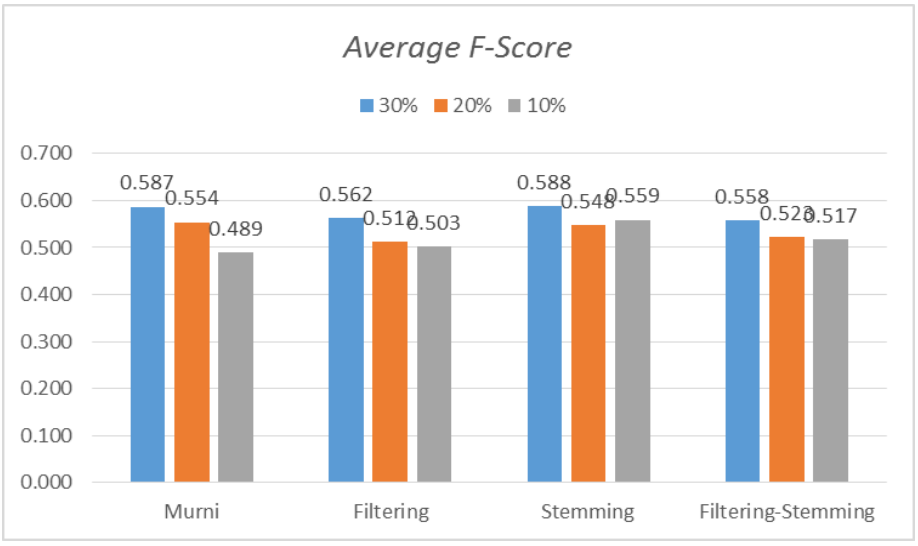

Figure 6 Average of non-semantic-based measurement f-scores with compression of 30%, 20%, and 10%

4. CONCLUSIONS

Based on the research and test results, it can be concluded as follows: The best f-score value on MMR method with semantic-based measurement (with the understanding of natural language) is 0.561, while with non-semantic-based measurements (without understanding natural language) is 0.588. This value is generated by adding a preprocessing process in the form of stemming and compression of 30% summary results.

REFERENCES

[1] A. Khan and N. Salim, "A Review on Abstractive Summarization Methods," *J. Theor. Appl. Inf. Technol.*, vol. 59, no. 1, pp. 64–72, 2014.

[2] N. Andhale and L. A. Bewoor, "An Overview of Text Summarization Techniques," *2016 Int. Conf. Comput. Commun. Control Autom.*, pp. 1–7, Aug. 2017.

[3] P. Krishnaveni and S. R. Balasundaram, "Automatic Text Summarization by Local Scoring and Ranking for Improving Coherence," *2017 Int. Conf. Comput. Methodol. Commun.*, pp. 59–64, 2017.

[4] M. Afsharizadeh, H. E. Komleh, and A. Bagheri, "Query-oriented Text Summarization using Sentence Extraction Technique," *2018 4th Int. Conf. Web Res.*, pp. 128–132, 2018.

[5] P. M. Sabuna and D. B. Setyohadi, "Summarizing Indonesian Text Automatically By Using Sentence Scoring And Decision Tree," *2017 2nd Int. Conf. Inf. Technol. Inf. Syst. Electr. Eng.*, pp. 1–6, 2017.

[6] P. P. Tardan, A. Erwin, and I. T. Faculty, "Automatic Text Summarization Based on S emantic Analysis Approach for Documents in Indonesian Language," *013 Int. Conf. Inf. Technol. Electr. Eng.*, pp. 1–6, 2013.

[7] R. Reztaputra and M. L. Khodra, "Sentence Structure-based Summarization for Indonesian News Articles," *2017 Int. Conf. Adv. Informatics, Concepts, Theory, Appl.*, pp. 0–5, 2017.

[8] D. Annisa and M. L. Khodra, "Query-based Summarization for Indonesian News Articles," *2017 Int. Conf. Adv. Informatics, Concepts, Theory, Appl.*, 2017.

[9] J. Carbonell and J. Goldstein, "The Use of MMR, Diversity-Based Reranking for Reordering Documents and Producing Summaries," in *Proceedings of the 21st annual*

international ACM SIGIR conference on Research and development in information retrieval - SIGIR '98, 1998, pp. 335–336.

[10] G. Yapinus, A. Erwin, M. Galinium, and W. Muliady, "Automatic Multi-Document Summarization for Indonesian Documents Using Hybrid Abstractive- Extractive Summarization Technique," *Inf. Technol. Electr. Eng. (ICITEE), 2014 6th Int. Conf.* , pp. 1–5, 2014.

[11] D. Cao and L. Xu, "Analysis of Complex Network Methods for Extractive Automatic Text Summarization," in *2016 2nd IEEE International Conference on Computer and Communications (ICCC)*, 2016, pp. 2749–2756.

[12] Y. Li, D. McLean, Z. A. Bandar, J. D. O'Shea, and K. Crockett, "Sentence Similarity Based on Semantic Nets and Corpus Statistics," *IEEE Trans. Knowl. Data Eng.*, vol. 18, no. 8, pp. 1138–1150, Aug. 2006.

[13] N. H. M. Noor, S. Sapuan, and F. Bond, "Creating the Open Wordnet Bahasa," *25th Pacific Asia Conf. Lang. Inf. Comput.*, pp. 255–264, 2011.

[14] A. Dinakaramani, F. Rashel, A. Luthfi, and R. Manurung, "Designing an Indonesian Part of Speech Tagset and Manually Tagged Indonesian Corpus," *Proc. Int. Conf. Asian Lang. Process. 2014, IALP 2014*, pp. 66–69, 2014.

[15] K. Shetty and J. S. Kallimani, "Automatic Extractive Text Summarization Using K-Means Clustering," *Electr. Electron. Commun. Comput. Optim. Tech. (ICEECCOT), 2017 Int. Conf.*, 2017.