Best Parameter Selection Of Rabin-Karp Algorithm In Detecting Document Similarity

Anggit Dwi Hartanto
Faculty of Computer Science
Universitas Amikom Yogyakarta
Yogyakarta, Indonesia
anggit@amikom.ac.id

Andy Syaputra
Faculty of Computer Science
Universitas Amikom Yogyakarta
Yogyakarta, Indonesia
andygrap@gmail.com

Yoga Pristyanto
Faculty of Computer Science
Universitas Amikom Yogyakarta
Yogyakarta, Indonesia
yoga.pristyanto@amikom.ac.id

***Abstract*— Text mining is usually used to detect document similarities and plagiarism. The field of education is one area that is prone to plagiarism. Plagiarism can kill someone's creativity because this action does not require energy and does not have to think hard. Therefore, the act of plagiarism must be prevented from causing harm to various parties. By using matching strings on documents, it can be used to detect plagiarism. One method that can be used is Rabin-Karp Algorithm, but in several studies that have been done the researchers did not test the k-gram value and database value, in theory, this would affect the performance of the Rabin-Karp Algorithm. Therefore in this study, the selection of k-gram values and prime bases was conducted to determine the effect on the performance of the Rabin-Karp Algorithm. The results showed that the selection of gram values and prime bases affected the processing time in testing the data and the similarity values of the documents being tested. In this study the value of k = 5 on k-gram has the fastest time for the testing process, both testing with multiple data 25 and testing the data for all amounts of data the number is 300.**

Keywords*— *Text Mining, Rabin-Karp, Parameter Selection, Document Similarity

I. Introduction

Artificial intelligence is one of the fields of currently popular computer science and is becoming a trend in the development of information technology. Artificial intelligence is exercised to help humans accomplish their jobs. One field of artificial intelligence that is prevalent today is text mining [1]. Text mining is the process of extracting patterns from a large number of unstructured data sources in the form of text or strings. One of the uses of text mining is to detect text similarities or plagiarism in documents [2]. The field of education is one area that is prone to plagiarism. One example is in making a thesis, there are often similarities between theses. Plagiarism can kill someone's creativity because this action does not require energy and does not have to think hard. Therefore, the act of plagiarism must be prevented from causing harm to various parties. By using matching strings on documents, it can be used to detect plagiarism [3].

Various studies have been conducted regarding the detection of plagiarism in a document, in the detection of plagiarism there are several algorithms that are often used but the popular and frequently used algorithm is Rabin-Karp. In a study conducted by Shivaji et al. [4] pplying Rabin-Karp Algorithm in detecting similarities in essay examination documents, the results of Rabin-Karp Algorithm were able to detect the similarity of the document based on string matching. Whereas Putri et al. [5] in her research used Rabin-Karp Algorithm to detect the similarity of the final assignment abstraction. In addition, Putera et al. [6] also conducted a study regarding the detection of similarities on online assignments using Rabin-Karp Algorithm, the results of Rabin-Karp Algorithm being able to detect the similarity of the document properly.

However, from several studies carried out related to the application of Rabin-Karp Algorithm, the majority of studies did not select k values for k-gram and the prime base in Rabin-Karp Algorithm. The k value of k-gram and the prime base has an influence on the performance of Rabin-Karp Algorithm, this is because k-gram and prime base calculation are an essential step and must be done on the application of Rabin-Karp Algorithm. Therefore research is needed regarding the selection of k values for k-gram and the optimal prime base in the application of Rabin-Karp Algorithm. This study aims to select k values for k-gram and prime base to determine k value for k-gram and optimal prime base in detecting document similarities using Rabin-Karp Algorithm. Contributions made in this study are :

1) The research that we propose can be a solution to find out whether the process of selecting k values for k-gram and prime base can increase the accuracy value of Rabin-Karp Algorithm or not.

2) This research can be a reference for further research related to the application of Rabin-Karp Algorithm to the detection of document similarities.

II. Methods

The following is the presentation of the research steps shown in Figure 1.

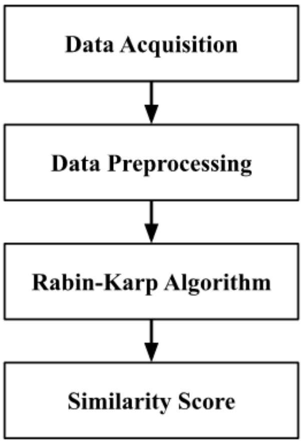

Figure 1. Research Flowchart

This following steps are the explanation for each of the process on the flowchart shown in Figure 1.

A. Data Acquisition

In this study, the data used is thesis abstract data at a university in the form of text. Data is obtained from a university's repository.

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE

978-1-7281-1655-6/19/$31.00 ©2019 IEEE

B. Data Preprocessing

The following is the presentation of the preprocessing steps shown in Figure 2.

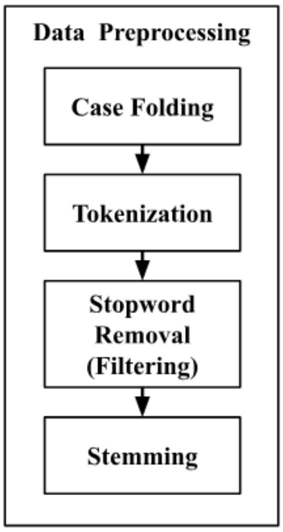

Figure 2. Preprocessing Step

This following steps are the explanation for each of the process on the flowchart shown in Figure 2.

1) Case Folding

Case folding is the process of converting all the letters in a document into lower case letters. Only the letters "a" to the letters "z" are received. Non-letter characters are omitted and are considered delimiters [7]. The following is Figure 3 example of case folding process in Indonesian text.

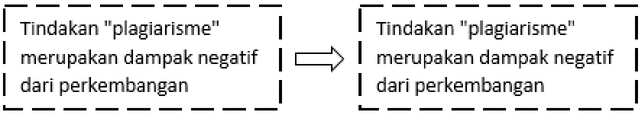
Tindakan "plagiarisme" merupakan dampak negatif dari perkembangan
Tindakan "plagiarisme" merupakan dampak negatif dari perkembangan

Figure 3. Case Folding Process

2) Tokenizing

Tokenizing has a function to cut the input string based on each word that composes it [8]. The following is Figure 4 example of tokenizing process in Indonesian text.

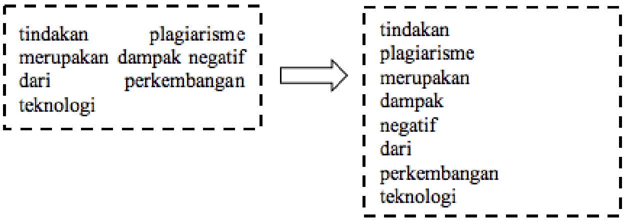
tindakan plagiarisme merupakan dampak negatif dari perkembangan teknologi
tindakan
plagiarisme
merupakan
dampak
negatif
dari
perkembangan
teknologi

Figure 4. Tokenizing Process

3) Stopword Removal (Filtering)

Filtering is the process of taking important words or having meaning only from the results of tokenizing. In this process, words that do not have meaning like conjunctions will be omitted [9]. In this process, the stopword list is usually used as a reference to eliminate words. The following is Figure 5 example of filtering process in Indonesian text.

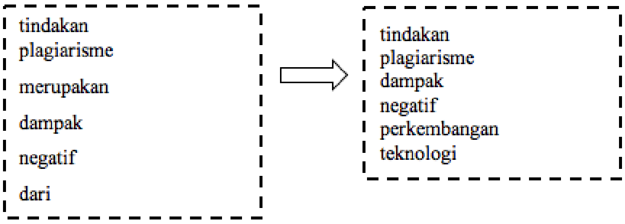

Figure 5. Filtering Process

4) Stemming

Stemming is one way that is used to improve the performance of information retrieval by transforming words in a text document into the basic word form, commonly referred to as root word. [10]. The following is Figure 6 example of stemming process in Indonesian text..

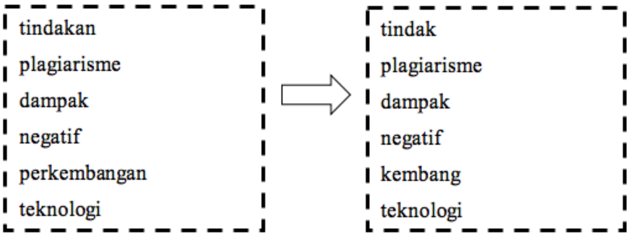

Figure 6. Stemming Process

In this study, the stemming algorithm used is the Nazief-Adriani Algorithm, because the Nazief-Adriani algorithm has a good ability to stemming Indonesian text. [11].

C. Rabin-Karp Algorithm

The following is the presentation of the Rabin-Karp algorithm steps shown in Figure 7.

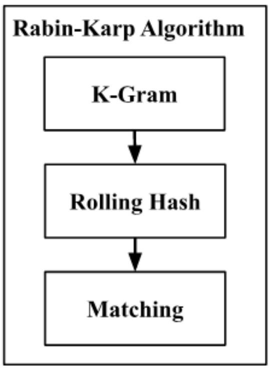

Figure 7. Rabin-Karp Algorithm

This following steps are the explanation for each of the process on the flowchart shown in Figure 7.

1) K-Gram

K-Gram is a series of terms with length k. The majority used as terms are words. K-Gram is used to extract letters of characters from a number of k from a word continuously read from the source text to the end of the document [6]. The following is an example of applying K-Gram with a value of k = 5 in Indonesian text:

- Indonesian text example : *"Sistem pendeteksi dugaan plagiat"*
- Then delete the space character : *"sistempendeteksidugaanplagiat"*

- Thus produced a series of 5-grams derived from the text as follows : *"siste istem stemp tempe empen mpend pende endet ndete detek eteks teksi eksid ksidu sidug iduga dugaa ugaan gaanp aanpl anpla nplag plagi lagia agiat"*

2) *Rolling Hash*

Rolling Hash or hashing is a stage that transforms a string into a fixed-length unique value that functions as a string marker. The function to generate this value is called the hash function, while the resulting value is called the hash value [12]. Here is the equation for calculating the hash value for Rabin-Karp Algorithm [13].

$$h(s)= (s[i] *b^{(n-1)} + s[i+1] *b^{(n-2)} +...+s[i+n-1]) \bmod q \qquad (2)$$

Where :

s = Search string length.

i = Array to i value.

b = Prima Base.

n = Number of string lengths searched.

q = Modulo.

3) *Matching*

Matching or string matching is a step to compare the hash value of input text to the hash value of the thesis abstract document in the repository [14].

D. *Similarity Score*

Calculation of Similarity Score in this study uses the Dice's Similarity Coefficient equation [15] as follows :

$$S = \frac{2 \times C}{A + B} \qquad (1)$$

Where :

S : Document similarity score.

A & B : Number of hash schemes in documents A and B.

C : The same hash number of two documents.

III. Results And Discussion

In this study, we conducted an examination of input documents in the form of thesis abstracts and compared with 4 thesis abstract documents randomly selected in the repository. In the experiments, we conducted we tested the similarity of documents using the Rabin-Karp Algorithm. The experiment was conducted by selecting the k-gram value and prime base value. These values will then be compared which will then choose the best value. The gram values that will be tested are 4, 5, and 6. While the prime base values to be tested are 5, 11, and 23. That value is chosen because the majority of studies use one of these values. The following is Table 1. The similarity value of the experimental results using the Rabin-Karp algorithm with the selection of k-gram values and prime base values.

TABLE I. Similarity Score

		Abstract 1	Abstract 2	Abstract 3	Abstract 4
4-gram	**5**	29.33	38.73	35.95	60.67
	11	30.33	37.96	35.31	57.71
	23	32.9	36.94	38.02	56.48
5-gram	**5**	27.36	0	34.32	58.63
	11	29.94	35.2	38.19	55.96
	23	29.51	34.18	35.23	54.72
6-gram	**5**	29.98	35.76	36.82	55.82
	11	28.12	34.48	38.11	57.73
	23	31.99	38.19	34.88	55.53

The following is Figure 8 is the similarity value of the experimental results using the Rabin-Karp algorithm with the selection of k-gram values and prime base values.

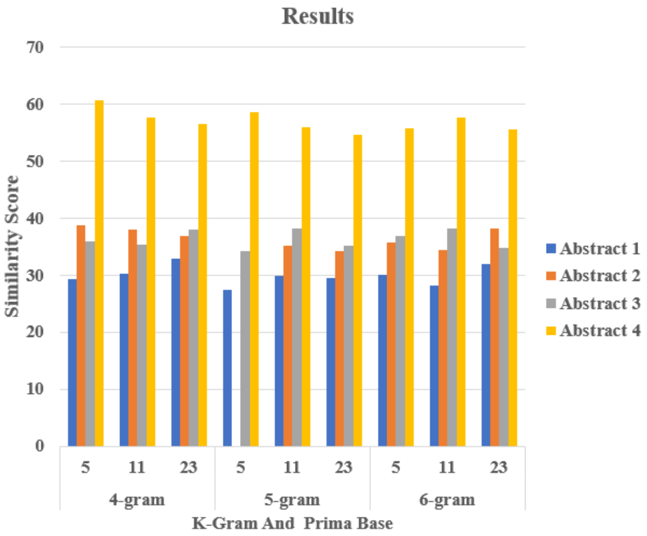

Figure 8. Results

Based on Table 1 and Figure 8 the selection of k values on k-gram and the value of prime base numbers by passing the stages of Stemming and Rolling Hash with modulo can give an effect on the similarity score produced by the Rabin-Karp Algorithm. This is because the size of the K-Gram value affects the breaking of the string in the form of K-Gram.

In addition, in this study, we also conducted a trial to measure process speed based on the gram-value of the Rabin-Karp algorithm. The k values for the grams we compare are 4.5, and 6. Here is Table 2. This is the processing speed based on the k-gram value of the Rabin-Karp algorithm.

TABLE II. TIME PROCESS

Number of Documents	Processing Time (Seconds)		
	4-gram	5-gram	6-gram
25	0.072	0.071	0.075
50	0.152	0.131	0.148
75	0.196	0.166	0.194
100	0.266	0.24	0.26
125	0.339	0.275	0.311
150	0.374	0.33	0.368
175	0.406	0.39	0.422
200	0.53	0.445	0.502
225	0.585	0.549	0.578
250	0.648	0.645	0.646
300	0.799	0.717	0.734

The following is Figure 9 is a comparison of processing time by the Rabin-Karp algorithm with a value of k on different k-grams..

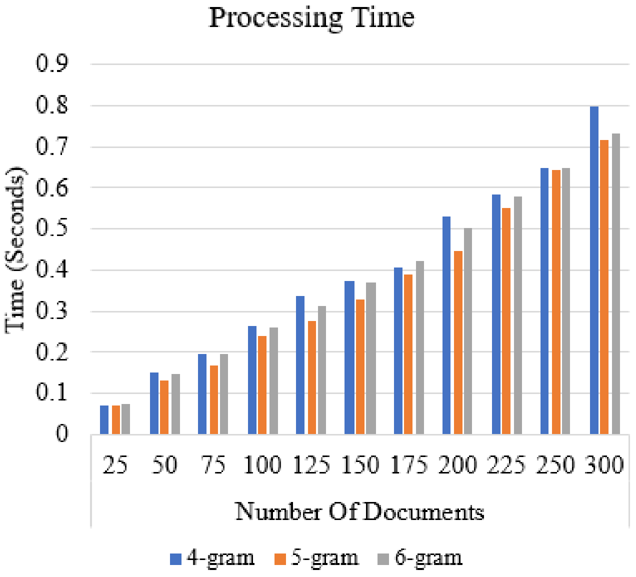

Figure 9. Comparison of ProcessingTime

Based on Table 2 and Figure 9, we can find the fastest time in the trial process for 300 data abstractions. With the time of each 4-Gram value is 0.799 seconds, the 5-Gram value is 0.717 seconds and the 6-Gram value is 0.734 seconds. And for the average results with a trial of multiple data 25, the K-Gram 4 value is 0.397 seconds, the value of K-Gram 5 is 0.360 seconds and the value of K-Gram 6 is 0.385 seconds.

IV. CONCLUSION

Selection of k value for k-gram and prime base value can influence the similarity score of Rabin-Karp Algorithm. However, the effect is not too significant, this can be seen from the experimental results which show the difference in the average similarity score between the k-gram values and the prime base value compared there is no significant difference. On the other hand, the results of testing processing time with different k-gram values indicate a difference, in this study the value of k = 5 on k-gram has the fastest time for the testing process, both testing with multiple data 25 and testing the data for all amounts of data the number is 300. Besides being influenced by the k value of k-gram processing speed is also influenced by the hardware used. For future research, it is suggested to try to optimize the stemming process by testing several other stemming algorithms.

ACKNOWLEDGMENT

Thanks to the Faculty of Computer Science, Universitas Amikom Yogyakarta who helped this research.

REFERENCES

[1] S. S. Hasan, F. Ahmed, and R. Surovi Khan, “Approximate String Matching Algorithms: A Brief Survey and Comparison,” *Int. J. Comput. Appl.*, vol. 120, no. 8, pp. 26–31, 2015.

[2] A. Parker and J. O. Hamblen, “Computer Algorithms for Plagiarism Detection,” *IEEE Trans. Educ.*, vol. 32, no. 2, pp. 94–99, 1989.

[3] T. Mardiana, T. Bharata Adji, and I. Hidayah, “Stemming Influence on Similarity Detection of Abstract Written in Indonesia,” *TELKOMNIKA (Telecommunication Comput. Electron. Control.*, vol. 14, no. 1, p. 219, 2016.

[4] S. K. Shivaji and P. S, “Plagiarism Detection by using Karp-Rabin and String Matching Algorithm Together,” *Int. J. Comput. Appl.*, vol. 115, no. 23, pp. 37–41, 2015.

[5] R. E. Putri, A. Putera, and U. Siahaan, “Examination of Document Similarity Using Rabin-Karp Algorithm,” *Int. J. Recent Trends Eng. Res.*, vol. 3, no. 8, pp. 196–201, 2017.

[6] S. Andysah Putera Utama, M. Mesran, R. Robbi, and S. Dodi, “K-Gram As A Determinant Of Plagiarism Level In Rabin-Karp Algorithm,” *Int. J. Sci. Technol. Res.*, vol. 6, no. 07, pp. 350–353, 2017.

[7] A. F. Hidayatullah, C. I. Ratnasari, and S. Wisnugroho, “Analysis of Stemming Influence on Indonesian Tweet Classification,” *TELKOMNIKA (Telecommunication Comput. Electron. Control.*, vol. 14, no. 2, p. 665, 2016.

[8] A. Yudhana, S. Sunardi, and A. Djalil, “Implementation of Pattern Matching Algorithm for Portable Document Format,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 11, pp. 509–512, 2017.

[9] D. A. Kurniawan, S. Wibirama, and N. A. Setiawan, “Real-Time traffic classification with Twitter data mining,” in *Proceedings of 2016 8th International Conference on Information Technology and Electrical Engineering: Empowering Technology for Better Future, ICITEE 2016*, 2017.

[10] P. M. Prihatini, I. K. G. D. Putra, I. A. D. Giriantari, and M. Sudarma, “Stemming Algorithm for Indonesian Digital News Text Processing,” *Int. J. Eng. Emerg. Technol.*, vol. 2, no. 2, pp. 1–7, 2017.

[11] H. R. Pramudita, “Penerapan Algoritma Stemming Nazief & Andriani dan Similarity Pada Penerimaan Judul Thesis,” *J. Ilm. DASI*, vol. 15, no. 04, pp. 15–19, 2014.

[12] A. P. U. Siahaan *et al.*, “Combination of Levenshtein Distance and Rabin-Karp to Improve the Accuracy of Document Equivalence Level,” *Int. J. Eng. Technol.*, vol. 7, no. 2.27, pp. 17–21, 2018.

[13] A. P. U. Siahaan, "Rabin-Karp Elaboration in Comparing Pattern Based on Hash Data," *Int. J. Secur. Its Appl.*, vol. 12, no. 2, pp. 59–66, 2018.

[14] A. Hamza Osman, N. Salim, and A. Abuobieda, "Survey of Text Plagiarism Detection," *Comput. Eng. Appl. J.*, vol. 1, no. 1, pp. 37–45, 2018.

[15] R. B. Aji, Z. A. Baisal, and Y. Firdaus, "Automatic Essay Grading System Menggunakan Metode Latent Semantic Analysis," *Semin. Nas. Apl. Teknol. Inf.*, vol. 2011, no. Snati, pp. 1–9, 2011.