Journal of Applied Intelligent System (e-ISSN : 2502-9401 | p-ISSN : 2503-0493)
Vol. 6 No. 1, 2021, pp. 10 – 22

Comparation of Dice Similarity and Jaccard Coefficience Against Winnowing Algorithm For Similarity Detection of Indonesian Text Documents

Santi Purwaningrum *[1], Agus Susanto [2]
Politeknik Negeri Cilacap, Jl. Dr. Soetomo No. 1, Sidakaya, (0282) 533329
E-mail : santi.purwaningrum@pnc.ac.id[1], agussusanto@pnc.ac.id [2]

Nur Wachid Adi Prasetya
Politeknik Negeri Cilacap, Jl. Dr. Soetomo No. 1, Sidakaya, (0282) 533329
E-mail : nwap.pnc@pnc.ac.id

Abstract - Plagiarism is the act of imitating and quoting or even copying of acknowledging other people's work as one's own work. The act of plagiarism currently has developed very fast especially in the world of education. Therefore, plagiarism detection is needed to prevent plagiarism from growing rapidly. This paper intends to conduct research on document similarity detection with the Winnowing algorithm which functions to find the fingerprint value in each document. After the winnowing process, the next step is to find the best value for the similarity level of each document by comparing the dice similarity and the jaccard coefficient. The test results show that the use of dice similarity is better with an average similarity value of 71.17615% compared to using the jaccard coefficient with a similarity value of 35.58837%.

Keywords -Plagiarism, Winnowing, Dice Similarity, Jaccard Coefficient

1. INTRODUCTION

The development of information technology in this globalization era is very fast, it also requires people to have all digital lifestyle. These developments will have positive or negative impacts; the positive impact for information technology will help us to find information for material reference and digital publication for someone's written work, while the negative impact allows someone's published work to be copied easily.

Imitating or plagiarism is the whole of taking the whole idea, concept other people's thought in writing, song, chat, discussion. Taking idea is directed to ideas have become work and written form, composition or other forms of expressions[1]. The act of plagiarism is usually called plagiarism and the people who do plagiarism are called plagiarists. Plagiarism can be found in academic environment, because students often interact with computer that have facility to copy the content of one document then paste it into another document. Computer facility and technology development sometimes are misused by students to do plagiarism in doing their final project or thesis.

Final project or thesis is one of the main requirements for completing the university level studies to get an intermediate or bachelor's degree. Many students think that final project or thesis is very difficult activity. So that many students deliberately cheating by committing plagiarism because they do not understand the lecture materials being taught, or students accidentally committing plagiarism due to a lack of knowledge about how to cite and include the source of information properly and correctly.

Based on the problem above, it needs a method that can be used to detect document similarity to reduce the plagiarism in doing the final project or thesis. Similarity detection is a way or an effort to

search the similarity in document, from the result of detection can be seen what percentage of the document similarities are compared[2]. Plagiarism detection system is divided into two systems, Intrinsic Plagiarism Detection (IPD) and External Plagiarism Detection (EPD). The work process of IPD system is only based on the imitation of human expertise in recognizing parts of the text that experience a change in writing style as a sign of copy or paste text without comparing with other text[3].

EPD system process compares each document inputted with each document contained in the corpus to compare *similarity* [4]. Corpus must have several documents that have the same topic with the source of plagiarism to know the test of document *similarity* level. One of algorithms included in the process of EPD is Winnowing Algorithm.

Winnowing Algorithm is developing from Rabin-Karp Algorithm. Developing winnowing algorithm to rabin-karp is on winnowing algorithm included window concept to increase the result of detection; on window process is substring formation along the k-gram[5]. Winnowing algorithm is used to detect words similarity in two documents. While rabin-karp algorithm, it is used to search for the number of the string [6].

According to N. Alamsyah [7] that winnowing algorithm, it has its own ways to find the similarity of word on the thesis tittle with *fingerprint.* The test is still the text form then it will be changed into a numeral which is called hash. Then the value of hash will be used to find the value on the thesis tittle being submitted. Then it is grouped for each value of hash called window. And the smallest number of each window called *fingerprint*. From this fingerprint, the lecturer will know whether the tittle of thesis being submitted by the student is plagiarism or not. The Similarity calculation using *Jaccard Coefficient.*

According to Nur Alamsyah's research to detect document similarity with *fingerprinting* method, it can be done by comparing algorithm related to the field of the tet mining, such as *winnowing*, *rabin karp* and *manber*. On his research, winnowing approach is better than rabin karp approach, because it produces the smaller and faster processing time with the 8th document trial, with value n-gram=9 and window = 3, time processing 0.02574 with the smallest similarity 32.6%[8]. This researh has also been conducted by Putra et al, in his research, it was used rabin karp algorithm to detect the similarity of two texts compared by transforming them into a series of number referring to ASCII table, it is also called the hashing process. The *Dice's similarity coefficient* application in counting the value of *similariry*, which is used K-gram approach[9]. Based on the background above, this research intends to conduct the level of document *similariry* contining the tittle and abstract of final proect comparing the *Dice similariry* and *jaccard coefficient* to find the best document similarity value level on winnowing algorithm which function to find the *fingerprint* value of each document.

2. RESEARCH METHOD

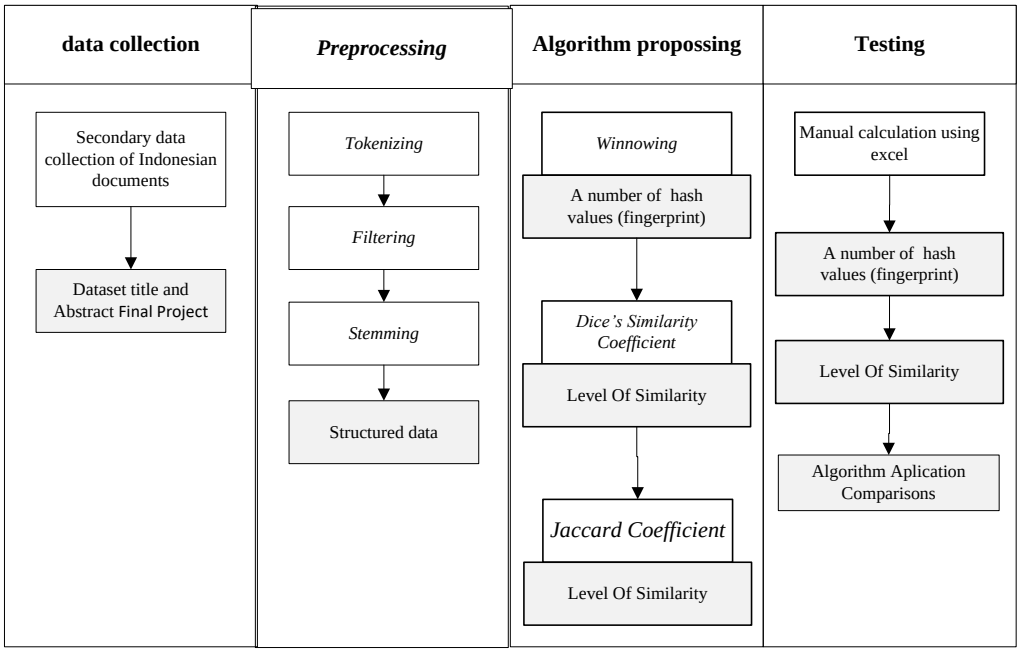


Figure 1. Research Step

In this research process, there are several stages were carried out, starting from where to obtain the data to research testing. The stages in this research are as follows:

2.1 Data collection

It is the first step in doing the research, how and where to obtain the data of the research. The data was obtained by collecting secondary data. Data sources are divided into two, namely primary data and secondary data. Secondary data may include data that has been previously gathered and is under consideration to be reused for new questions, for which the data gathered was not originally intended[10]. whereas employing secondary data is very useful, it also comes along with some serious (but manageable) caveats [11]. The dataset used in this research is it was the data of document containing abstract and tittle of the students of Politeknik Negeri Cilacap majoring Electrical Engineering. The dataset is a control system theme using Arduino which is divided into 5 files based on the type of use.

2.2 Preprocessing

Preprocessing is the first step of text mining process which functions to convert unstructured text data into structured text data. In this preprocessing process, it will do the steps to delete unimportant parts of the text in the document because it will be noise in the next process. The preprocessing steps are described below:

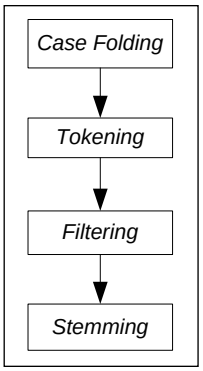


Figure 2. Preprocessing Process

a. *Case Folding*

It is the process to change all capital letters into small letter. In addition, it also removes all punctuation marks such as numbers, symbols, and others because it does not have unique values and it is not related to the string to be processed.

b. *Tokening*

It is the process to separate every word in the document. Example:
The first sentence: Ibu membeli 1 gaun digunakan untuk acara pesta.
Result:

ibu	membeli	gaun	digunakan
untuk	acara	pesta	

c. *Filtering*

The function of filtering is to throw away the meaningless words. The meaningless words are usually called stopword, the stopword such as: juga, dan, untuk, adalah etc. Example:
Sentence : ibu membeli gaun digunakan untuk acara pesta
Result : ibu membeli gaun digunakan acara pesta

d. *Stemming*

The function of stemming is to erase the affix of word in text document, so the words are taken is the basic words (root) to use in the next process. Example: mem-, -kan, ber-, -pun, me-an etc. Example:
Sentence : ibu membeli gaun digunakan acara pesta
Result : ibu beli gaun guna acara pesta

2.3 The proposed algorithm

In this research, we use the winnowing algorithm to find the fingerprit value in a document. Then find the best similarity value using the similarity dice and the jaccard coefficient.

1. Winnowing

Winnowing Algorithm is one of algorithm which functions as fingerprint document or algorithm to detect the act of plagiarism using hashing technique [12] [13]. The input from winnowing algorithm is a text document starting from preprocessing process then the output is a

number of Hash value. The hash value is a numeric value that is formed by the ASCII table calculation of each character. The hash value can also be referred to as a fingerprint, which is used as an indicator to compare the similarity of each document text.

The parameters of winnowing algorithm are *k-gram, hash,* and *window.* The explanation of winnowing algorithm generally as follows:

a. To eliminate punctuation mark and useless characters using preprocessing process in the first process because it will be noise in the next process.
b. To form a series of text *k-gram* that is still a number of strings. *K-gram* is a series of adjacent *substrings* of length [14]. K is a parameter determined by the user. A number of strings will be grouped into a new set of strings where it is a combination between the initial strings and the length strings which is the combination is K.
c. To do rolling hash process, it is used to get hash value from the series of *gram* that are formed. The changing of character series into a value or code then became a sign of the series of character, it is called *hashing* and the resulting value can be referred to as a hash value. This process uses the formula, as follows:

$$H(c_1..c_l) = c_1.b^{(l-1)} + c_2.b^{(l-2)} + .. + c_{(l-1)}.b + c_1 \qquad (1)$$

Where c is ASCII value of each character, l The length of string and b is User-defined base hash value.

e. To create a window with the result of the hash value of each previous gram. The window size is also determined by the user.
f. To determine the hash value of each window to be used as a document *fingerprint* and if there is the same hash value, the rightmost hash value will be selected

2. Dice Similarity

Dice Similarity is a method to calculatet the level of similarity between two objects [15]. The process of Dice similarity is to compare two documents by calculating the value of *k-gram*. Then the same number of *k-gram* of two documents are got *fingerprint* document. To calculate the value of similarity, it uses the following formula: [16]

$$S = \frac{2C}{A+B} x100 \qquad (2)$$

Where S is The value of Similarity, C A number of fingerprint from each documents are compared and A, B a number of fingerprint from each documents.

3. *Jaccard Coefficient*.

Jaccard coeffient is a set of measurement of similarity mostly applied on *Retrieval Information*, *Data Mining*, *Machine Learning* and many more [17]. In general, the calculation on accard Coefficient method based on *vector space similarity measure.* From each documents will be calculated the same words from one document to other documents, so it will produce the value of document similarity. If the value of similarity is higher, it means the document has many similarities. Jaccard coefficient calculates similarity between two obects A and B in two vectors [18].

$$Jaccard\ Coefficient\ (A,B) = \frac{|A \cap B|}{|(A \cup B)|} x100 \qquad (3)$$

On calculations *Jaccard coefficient,* the values (A, B) are The value of similarity between documents, | A∩B | is the A number of the same fingerprint from document 1 to other documents, | A∪B | is the A number of fingerprint value from document 1 and 2.

2.4 Testing

The purpose of this testing is comparing the searh of level similarity document between *dice similarity* and *jaccard coefficient* to winnowing algorithm. The test is conducted by searching the best value of *k-gram*, *window* and *hash* to detect case study of document containing the tittle and abstract of final project of the students of Politeknik Negeri Cilacap majoring Electrical Engineering. The testing process as follows:

1. Collecting the document containing of the tittle and abstracting 32 documents of Electrical Engineering' final project of Politeknik Negeri Cilacap's students in indonesia language, then dividing them into 6 groups based on the type of the use.
2. Determining the training data used as testing as the best parameter setting value (k-gram, hash, window). Training data takes one document source of plagiarism from each group type of using.
3. Calculating and comparing the level value of document similarity between dice similarity and jaccard coefficient on winnowing algorithm.

3. RESULTS AND DISCUSSION

Winnowing algorithm will be used to search fingerpint value then comapre with dice similarity and jaccard coefficient to find the best value of similarity on the data research. The number of data is 32 documents containing the tittle and abstract in indonesian languange of politeknik negeri cilacap's students majoring electrical engineering. Then divided into 6 based on the type of use.

Table 1. Dataset Using Detail

Type of data	Training	Testing
Documents about sorting or sorting an object	1	3
Monitoring of natural influences or phenomena	1	6
Animal feeding tool	1	2
Security process or security system	1	4
Simple robotic	1	8
Vehicle parking system	1	3

The result of fingerprint value of winnowing algorithm will be different if the setting of *k-gram, hash,* and *window* value are different. Therefore, it needs training process to determine k-gram, hash, and window value in accordance with maximal similarity result. The parameter setting proposed as follows:

Table 2. The setting value of k-gram, hash, and window are used in testing

K-gram	Hash	window
2	2	2

5	7	2
7	2	5
2	5	7
2	7	5
7	5	2
5	2	7

3.1 Fingerprint Searching

In this step will searh fingerprint value from each documents using winnowing algorithm. Data testing used is document of final project of the students containing the tittle and abstract, it will be compared from each documents based on the utility field of the tool. It concludes that system concepts of document similarity used winnowing algorithm as follows:

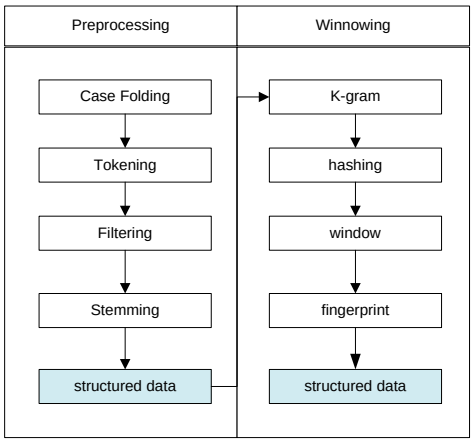


Figure 3. Document similarity concept using winnowing algorithm

The first result of document similarity detection is the result text from the preprocessing process. Example as follows:

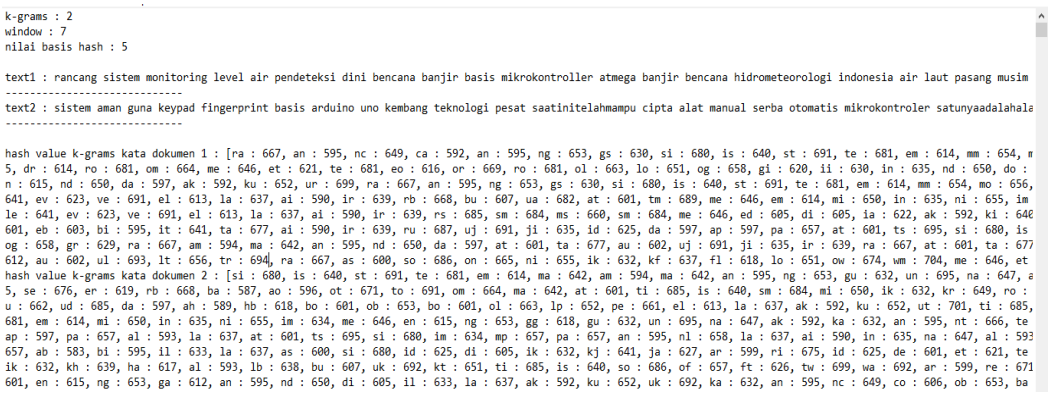
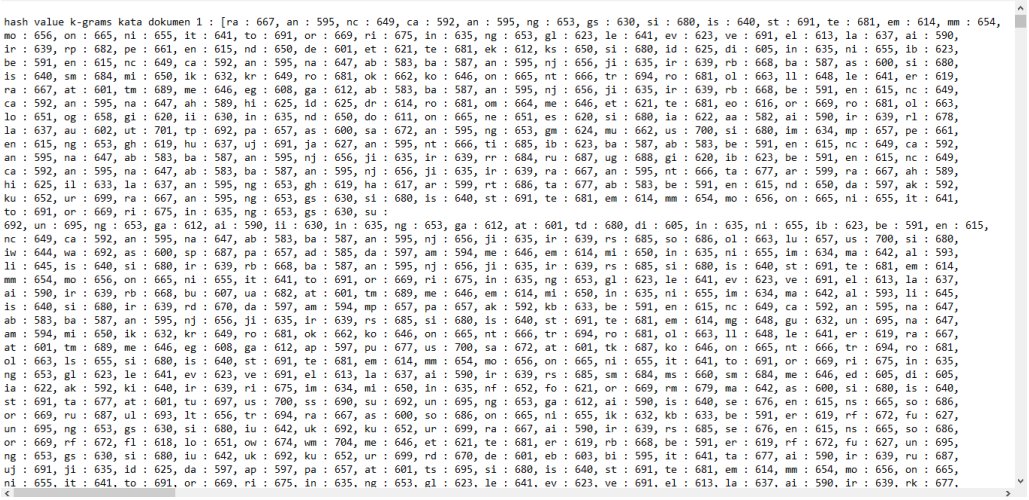


Figure 4. Document text after preprocessing

After the preprocessing process, the next process is the winnowing algorithm process. The first step in the winnowing process is to form the prepocessing result string into a series of k-grams. The old string set is grouped into a new set of strings, the result of the new string is the result of concatenating the old strings with the length of the string concatenated by k. example of k-gram results in document 1 as below with length k = 2 and hash = 5.



Figuere 5. The example of result of k-gram process and hashing

The following is the example of calculating of hash value above:
H ("ra") =
(114x (5 (2-1))) + (97x (5 (2-2)))
= 570 +97
= 667

After getting the result of hash value, the next step is forming window. Forming window process is like forming gram but it used the result of hash value. Example of specified windo size = 7.

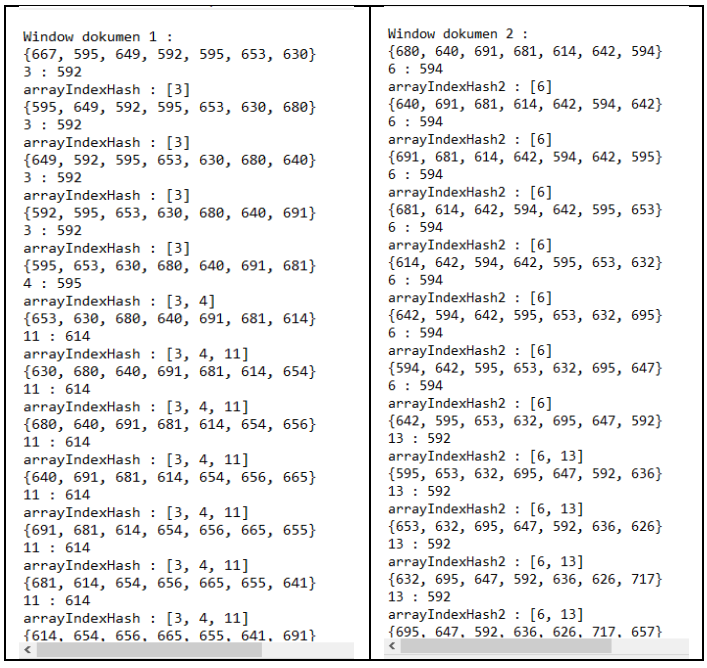


Figure 6. Window Result

The last step is choosing the smallest hash value from window which will be as a fingerprint. From the hash value specified in the window above, the minimum value that that will be used as a fingerprint is as follows:

```
Fingerprint dok 1 : [592, 595, 614, 641, 635, 623, 613, 590, 601, 605, 591, 583, 587, 600, 632, 646, 619, 589, 616, 620, 611, 582,
602, 624, 615, 630, 612, 639, 644, 585, 594, 593, 597, 655, 640, 634, 621, 656, 627, 618, 599, 617, 530, 339, 607]
Fingerprint dok 2 : [594, 592, 585, 602, 615, 616, 619, 587, 599, 617, 635, 614, 595, 612, 620, 601, 582, 613, 589, 600, 593, 596,
632, 646, 641, 644, 597, 634, 607, 640, 609, 605, 672, 590, 583, 625, 606, 588, 626, 630, 598]
Fingerprint dok yang sama : [592, 595, 614, 641, 635, 613, 590, 601, 605, 583, 587, 600, 632, 646, 619, 589, 616, 620, 582, 602,
615, 630, 612, 644, 585, 594, 593, 597, 640, 634, 599, 617, 607]
------------------------- Dokumen 1 vs Dokumen 2 -------------------------
```

Figure 7. Fingerprint Results

3.2 Measuring Similarity Values

In this step, the similarity value between documents will be calculated. If the fingerprint has been obtained from each document using the winnowing algorithm, then the next step is to compare determining the similarity value using Dice's similarity and Jaccard coefficient.

3.2.1 *Dice Similarity*

It is the example of calculating dice similarity using fingerprint value from the result of winnowing process previously.

A number of fingerprint document 1 = 45
A number of fingerprint document 2 = 41
A number of the same fingerprint document = 33

So: $S = \frac{2x33}{45+41} x100$

$$S = \frac{66}{86} = 76.74419$$

```
jumlah hash yg sama = 33
Hash k-grams kata yang sama = 592
Hash k-grams kata yang sama = 595
Hash k-grams kata yang sama = 614
Hash k-grams kata yang sama = 641
Hash k-grams kata yang sama = 635
Hash k-grams kata yang sama = 613
Hash k-grams kata yang sama = 590
Hash k-grams kata yang sama = 601
Hash k-grams kata yang sama = 605
Hash k-grams kata yang sama = 583
Hash k-grams kata yang sama = 587
Hash k-grams kata yang sama = 600
Hash k-grams kata yang sama = 632
Hash k-grams kata yang sama = 646
Hash k-grams kata yang sama = 619
Hash k-grams kata yang sama = 589
Hash k-grams kata yang sama = 616
Hash k-grams kata yang sama = 620
Hash k-grams kata yang sama = 582
Hash k-grams kata yang sama = 602
Hash k-grams kata yang sama = 615
Hash k-grams kata yang sama = 630
Hash k-grams kata yang sama = 612
Hash k-grams kata yang sama = 644
Hash k-grams kata yang sama = 585
Hash k-grams kata yang sama = 594
Hash k-grams kata yang sama = 593
Hash k-grams kata yang sama = 597
Hash k-grams kata yang sama = 640
Hash k-grams kata yang sama = 634
Hash k-grams kata yang sama = 599
Hash k-grams kata yang sama = 617
Hash k-grams kata yang sama = 607
Similarity Dice Similarity = 76.74418687820435 %
```

Figure 8. The Result of Similarity Dice Similarity

3.2.2 *Jaccard Coefficient*

The input is the same as the dice similarity process, using fingerprint value from the previous winnowing process but using the following formula:

$$S = \frac{33}{45+41} x100 = 38.37209$$

```
jumlah hash yg sama = 33
Hash k-grams kata yang sama = 592
Hash k-grams kata yang sama = 595
Hash k-grams kata yang sama = 614
Hash k-grams kata yang sama = 641
Hash k-grams kata yang sama = 635
Hash k-grams kata yang sama = 613
Hash k-grams kata yang sama = 590
Hash k-grams kata yang sama = 601
Hash k-grams kata yang sama = 605
Hash k-grams kata yang sama = 583
Hash k-grams kata yang sama = 587
Hash k-grams kata yang sama = 600
Hash k-grams kata yang sama = 632
Hash k-grams kata yang sama = 646
Hash k-grams kata yang sama = 619
Hash k-grams kata yang sama = 589
Hash k-grams kata yang sama = 616
Hash k-grams kata yang sama = 620
Hash k-grams kata yang sama = 582
Hash k-grams kata yang sama = 602
Hash k-grams kata yang sama = 615
Hash k-grams kata yang sama = 630
Hash k-grams kata yang sama = 612
Hash k-grams kata yang sama = 644
Hash k-grams kata yang sama = 585
Hash k-grams kata yang sama = 594
Hash k-grams kata yang sama = 593
Hash k-grams kata yang sama = 597
Hash k-grams kata yang sama = 640
Hash k-grams kata yang sama = 634
Hash k-grams kata yang sama = 599
Hash k-grams kata yang sama = 617
Hash k-grams kata yang sama = 607
Similarity Jaccard Coefficient = 38.37209302 %
```

Figure 9. Similarity Dice Similarity Results

3.3 Comparative test results of dice similarity and Jaccard coefficient on winnowing

In next step is winnowing algorithm testing step using the parameter setting values that have been selected based on table 2 which uses the training data listed in table 1. Then the results of the best parameter setting values will be used to search fingerprint then compare the search for similarity levels between and Jaccard coefficient using testing data.

3.3.1 Training step

The purpose of this step is to find the best setting value of K-gram, hash, and window that was gotten from testing using training data. Training testing was using 7 trials of parameter setting which will be tested with each training data. The total data obtained from the result of training testing is as many as 105 processes (7x15). The best parameter setting will be used for testing the data testing. The result of the best parameter setting values include the smallest value of the difference between parameters (difference in the results of similarity based on parameters) which is obtained from the results of the difference between dice similarity and jaccard coefficient, and the largest similarity value from the results of the similarity of dice similarity and jaccard coefficient.

Table 3. Parameter Testing Result

Parameter (K, H, W)	Dice Similarity (s)	Jaccard coeffiesn (%)	Difference Between Dice Similarity and Jaccard Coeffiesn (%)	Difference Between Parameters
2, 2, 2	87.1362	43,56833	43,56787	10.53477
2,5,7	66.06667	33.03357	33.0331	0.74743
2, 7, 5	64.57201	32.28634	32,28567	12.49333
5, 2, 7	39,58527	19.79292	19.79234	11,76494
7, 2.5	16.05533	8.027933	8,0274	4.709246
5, 7, 2	6.240553	2.9224	3.318154	2.27675

7, 5, 2	2.126533	1.085129	1.041404	1.041404

From the table above, it can be concluded that if the value of k-gram is higher, it will affect the result of similarity value and if the value of hash or window is higher, the similarity result will not be too big but significant enough to affect the difference between parameters.

From the parameter testing of the training data, the parameter value of k-gram = 2, hash = 5, and window = 7 are obtained then it will be used as data testing parameter to know the comparison of the calculation of document similarity using the similarity dice similarity and jaccard coefficient.

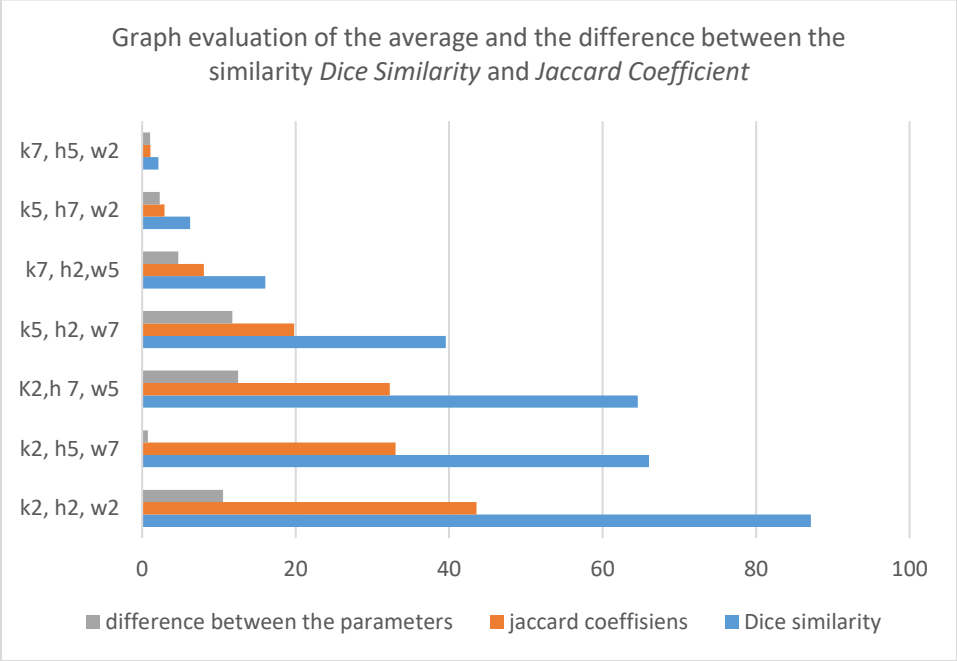


Figure 10. Evaluation Diagram of Average and Similarity Difference of Similarity Dice Similarity and Jaccard Coefficient

3.3.2 Testing Step

In this step, data testing will be carried out based on the parameter settings that have been obtained based table 3 and the parameter settings and total testing data listed in table 1 are 26 documents. So that, the process in the testing step as many as 110 processes to determine the difference in the level of similarity between documents.

Table 4. Evaluation Result of Testing the Difference in Level of Similarity and Jaccard Coefficient

Type of Data	Dice Similarity (%)	Jaccard coeffiesn (%)	Difference
Documents about sorting or sorting an object	68,278	34,13915	34.13885
Monitoring of natural influences or phenomena	69.04367	34,52216	34.5215
Animal feeding tool	69.23	34.61538	34,61462
Security process or security system	75,137	37,56884	37,56816
Simple robotic	70.33925	35.1699	35,16935
Vehicle parking system	75,029	37.51479	37,51421

Table 4 shows that the average of dice similarity 71.17615% and Jaccard Coefficient 35.58837%.

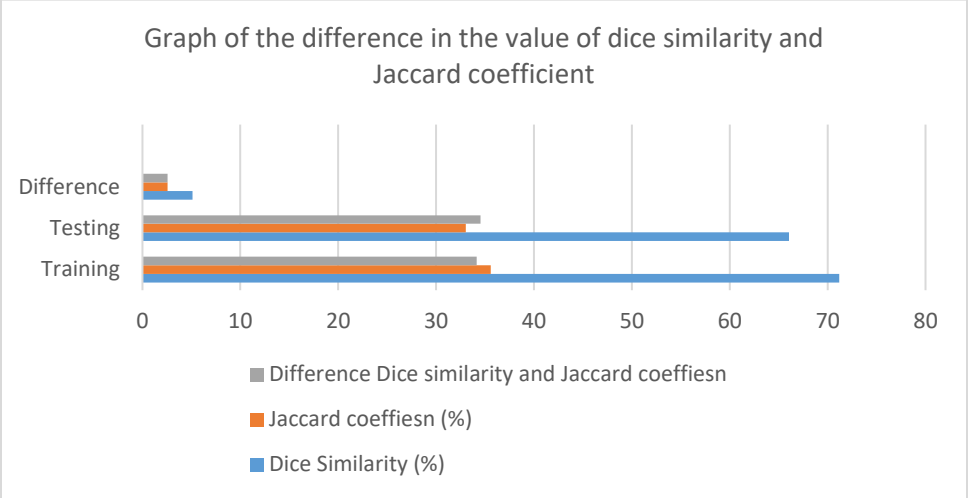


Figure 11. The difference between Dice Similarity and Jaccard Coefficient measurement test result

The result of the similarity level value using parameter setting at the training and testing step are not far away. The result of training and testing show that the cosine similarity has a higher similarity value than the Jaccard coefficient.

4. CONCLUSION

After testing the research, it can be concluded that: The best parameter setting is based on the smallest value of the difference between parameters (difference in the result of similarity basedon parameters) and the largest similarity value is from the result of the similarity of dice similarity and jaccard coefficient. The best result of setting research parameter with k-gram = 2, hash = 5, window = 7.

The result of parameter setting test shows that the higher k-gram value will affect the result of similarity value. And if the hash or window value is higher, so the changing of similarity result is not too big but significant enough to affect the difference between parameters.

Testing the value of the level of similarity using dice similarity to the winnowing algorithm is higher than the jaccard coefficient with the difference between the dice similarity and jaccard coefficient is 2.554683%.

This study is still have many weaknesses, so it is hoped that furthur researcher can correct it. The weaknesses need to be corrected such as typography error because in the writing of the final project is possible to make mistake in writing.

REFERENCES

[1] Stephen Fishman, *Public Domain: How To Fid & Use Copyright-Free Writings, Music, Art & More*, 4th ed. Berkeley: nolo, 2008.

[2] B. Sari and Y. Sibaroni, “Deteksi Kemiripan Dokumen Bahasa,” vol. 4, pp. 87–98, 2019, doi: 10.21108/indojc.2019.4.3.365.

[3] D. K. Bhattacharyya, “Plagiarism : Taxonomy , Tools and Detection Techniques Plagiarism and Its

Types," 2016.

[4] N. C. Haryanto, L. D. Krisnawati, and A. R. Chrismanto, "Temu Kembali Dokumen Sumber Rujukan dalam Sistem Daur Ulang Teks," *Jurnal Teknologi dan Sistem Komputer*, vol. 8, no. 2. pp. 140–149, 2020.

[5] R. K. Wibowo and K. Hastuti, "PENERAPAN ALGORITMA WINNOWING UNTUK MENDETEKSI KEMIRIPAN TEKS PADA TUGAS AKHIR MAHASISWA," *Techno.COM*, vol. 15, no. 4, pp. 303–311, 2016.

[6] L. Sibarani, M. Magdalena, and A. Dharma, "Analisa Perbandingan Sistem Pendeteksian Kemiripan Judul Skripsi Menggunakan Algoritma Winnowing Dan Algoritma Rabin Karp," *REMIK (Riset dan E-Jurnal Manaj. Inform. Komputer)*, vol. 4, no. 1, p. 69, 2019, doi: 10.33395/remik.v4i1.10174.

[7] N. Alamsyah *et al.*, "Deteksi plagiarisme tingkat kemiripan judul skripsi pada fakultas teknologi informasi menggunakan algoritma winnowing," vol. 10, no. 4, pp. 197–201, 2019.

[8] N. Alamsyah, "Perbandingan Algoritma Winnowing Dengan Algoritma Rabin Karp Untuk Mendeteksi Plagiarisme Pada Kemiripan Teks Judul Skripsi," *Technol. J. Ilm.*, vol. 8, no. 3, p. 124, 2017, doi: 10.31602/tji.v8i3.1116.

[9] N. P. Putra and Sularno, "Penerapan Algoritma Rabin-Karp Dengan Pendekatan Synonym Recognition Sebagai Antisipasi Plagiarisme Pada Penulisan Skripsi," *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 1, no. 2, pp. 49–58, 2019.

[10] TP Vartanian, *Secondary data analysis*. Oxford University Press: Oxford University Press, 2010.

[11] F. S. Martins, J. A. C. da Cunha, and F. A. R. Serra, "Secondary Data in Research – Uses and Opportunities," *Pod. Sport. Leis. Tour. Rev.*, vol. 7, no. 3, pp. I–IV, 2018, doi: 10.5585/podium.v7i3.316.

[12] S. Sunardi, A. Yudhana, and I. A. Mukaromah, "Indonesia Words Detection Using Fingerprint Winnowing Algorithm," *J. Inform.*, vol. 13, no. 1, p. 7, 2019, doi: 10.26555/jifo.v13i1.a8452.

[13] S. Sunardi, A. Yudhana, and I. A. Mukaromah, "Implementasi Deteksi Plagiarisme Menggunakan Metode N-Gram Dan Jaccard Similarity Terhadap Algoritma Winnowing," *Transmisi*, vol. 20, no. 3, p. 105, 2018, doi: 10.14710/transmisi.20.3.105-110.

[14] M. Y. Soleh and A. Purwarianti, "A Non Word Error Spell Checker for Indonesian using Morphologically Analyzer and HMM," no. July, 2011.

[15] F. Amin and E. Winarno, "Rancang Bangun Sistem Temu Kembali Informasi (Information Retrieval System) Dokumen Berbahasa Jawa menggunakan Metode DICE Similarity," vol. 21, no. 2, pp. 99–106, 2016.

[16] D. Gupta and V. K, "Investigating the Impact of Combined Similarity Metrics and POS tagging in Extrinsic Text Plagiarism Detection System Vani," pp. 1578–1584, 2015.

[17] J. Evan Harya Chandra, V. Christiani M, and D. S.Naga, "Plagiarisme Abstrak Menggunakan Algoritma Winnowing dan Synsets," *J. Ilmu Komput. dan Sist. Inf.*, pp. 121–129, 2016.

[18] L. J. Purba and L. Sitorus, "Perancangan Aplikasi Untuk Menghitung Persentase Kemiripan Proposal Dan Isi Skripsi Dengan Algoritma Rabin-Karp," *J. Tek. Inform. Unika St. Thomas*, vol. 3, no. 1, pp. 17–25, 2018.