Similarity detection design using Winnowing Algorithm as an effort to apply green computing

A Pratomo[1], A Irawan[1], M Risa[2]

[1] Department of Information Management, Politeknik Negeri Banjarmasin, Jalan Brig. Jend. Hasan Basri, Banjarmasin, Indonesia
[2] Department of Business Administration, Politeknik Negeri Banjarmasin, Jalan Brig. Jend. Hasan Basri, Banjarmasin, Indonesia

E-mail: adipratomo@poliban.ac.id

Abstract. The website is a media that is currently widely used as a tool to provide services and information. In this study, the application of a Winnowing Algorithm to detect the similarity of proposal documents through the website as an effort to apply green computing on campus. With the development of similarity detection is expected to use paper, computer resources will be reduced. Some test scenarios are carried out by changing the values of the gram and window parameters to obtain optimal values. This optimal value will be applied to the application to be built. The results of the implementation of the Winnowing Algorithm on the proposal management information system in the form of a plagiarism checking feature that is part of the system. The plagiarism checking feature will be able to produce output in the form of a percentage of the plagiarism rate that will be used as a recommendation for the department as an effort to apply green computing.

1. Introduction

The development of information technology today has developed rapidly, one of which is in terms of information management. By utilizing information technology support, current information processing can be done quickly, and can be presented globally by utilizing the internet network. Website is a medium that is currently widely used as a tool to provide services and information. In this era, information is easy to get and influence people. One of the negative impacts is the great number of plagiarism [1].

The Department of Business Administration currently uses the website for various purposes, including managing the submission of user proposals. At present the process of submitting a proposal can only be limited to the registration of proposal data on the website. After the proposal is registered, users still collect printed files. The file collection is intended so that the supervisor can carry out a process of examining the proposal. Examinations carried out include the contents and the level of similarity of the text with other proposals. The similarity of documents collected by students makes lecturers unable to work effectively and efficiently because the possibility of examining the same or similar assignments is very large [2]. This process is less efficient and less effective, but it also consumes resources that are not small, both human and computer resources. In addition, the process carried out so far also spent a lot of paper resources to print proposals. If each proposal has 25 pages and there are 200 proposals printed in 3 copies, then at least 15,000 sheets of paper will be needed. If this activity occurs

in one department, imagine how much paper is wasted in a single campus environment. After the proposal submission activity is completed, the proposal will be unused and wasted. This will certainly produce a lot of waste.

Based on the initial analysis of the system that is already running, it still does not have a system to detect the level of similarity, both for submitting titles and proposals. Similarity system is necessary for early detection of the similarity of documents submitted. Based on an analysis of the implementation over the past few years, it was found that several documents had a fairly high level of similarity. However, due to the absence of a detection system at the beginning of the system registration, the similarity of the documents submitted cannot be detected quickly.

In the academic world, plagiarism is a very avoidable thing. The students consider that conducting plagiarism is the fastest way to accomplish the assignment [3]. Therefore, to avoid plagiarism we need a system that can detect it early. In order to build this application, we need an algorithm that is able to detect the presence of the same sentence from the document being compared. In addition, the existence of this system will certainly be able to save human resources, computers and infrastructure advice. Utilization of a web-based plagiarism detection system can be used as a socialization tool and campaign to care for the environment and arouse the enthusiasm of the academic community to participate in using it as part of the spirit of green computing on campus. In addition, the existence of this system will certainly be able to save human resources, computers and paper in an effort to support Green Computing. Green computing is the behaviour of using computing resources efficiently, by maximizing energy, extending hardware life, minimizing paper usage, and several other technical matters. The main targets of green computing are the earth, humans, and profits. Green computing is ecologically sustainable computing. Green Computing works for saving the environment of computers, servers and associated devices such as a monitor, printers and networking and communications systems [4].

From several literature studies derived from studies with similar topics, explained that the plagiarism detection algorithm must meet three requirements, namely whitespace insensitivity, noise suppression, and position independence [2]. The Winnowing Algorithm satisfies all three of these things. The Winnowing Algorithm creates a fingerprint from a document through the three required processes. Approach to document clustering based on winnowing fingerprints that achieved good values of effectiveness with considerable save in memory space and computation time [5]. In this study, several scenarios were conducted to test the ability of the Winnowing Algorithm. The test scenario will be different in terms of parameter values. From this test, it is expected that the optimal value of parameters will be applied to the application to be built.

2. Methodology

This research uses design methods and system development methods. The method used is software lifecycle development by adopting the waterfall model. Waterfall model is recursive in that each phase can be endlessly repeated until it is perfected [6].

2.1. System requirement analysis system

Analysis stage is needed in the initial stages of system design and development to determine the needs of the detection system to be built. At this stage, a literature study related to retrieval of information system and fingerprint document method uses a material winnowing algorithm to identify the similarity of documents in the form of text.

2.1.1. Winnowing Algorithm as document similarity search. The results of the algorithm are based on the stage of accuracy and speed of process time by referring to previous studies written in the bibliography [7]. The stages of the winnowing algorithm process implemented in the system are explained in Figure 1 below.

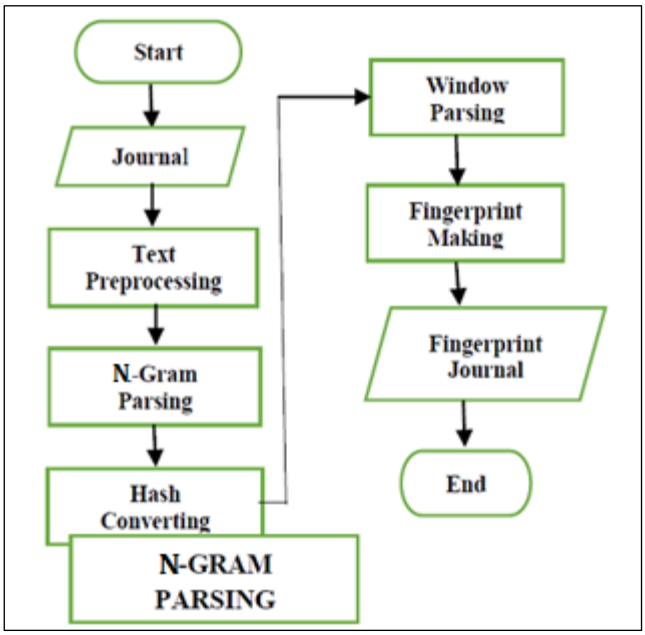

Figure 1. Stage of the Winnowing Algorithm process [5].

Winnowing algorithm is an algorithm used to make fingerprints from a document. The Winnowing algorithm converts text documents into a set of hash values called fingerprints. There are some basic needs that are used by winnowing algorithm in detecting the document similarity. The basic needs which have to be accomplished by detection algorithm are [8, 9]:

a. Whitespace insensitivity, it is a word search which is not influenced by space, punctuation, type of letter (capital or normal), etc.
b. Noise suppression, the function is to avoid the finding of short word and not the common word such as "the".
c. Position independence, it is a similarity finding that does not have to depend on the word position, so words with different order still can be recognized if there is a similarity.

The analysis technique of plagiarism is fingerprint analysis to create fingerprint document according to n-gram value that has been determined, then the similarity value will be counted according to the same number of fingerprint between texts [1]. The process for generating fingerprints from a document is as follows:

a. Discard irrelevant characters such as spaces, punctuation or articles or non-essential words, such as conjunctions. This step is in accordance with the requirements of the plagiarism detection algorithm, namely whitespace insensitivity and noise suppression.
b. Forms a series of n-grams from text. Such as determined by n = 6, it will produce a set of grams or strings measuring 6.
c. Performs a hash function for every gram. Equation (1) is a calculation of the hash function of the winnowing algorithm [4].

$$H\ c1\ldots ck = c1*bk-1+c2*bk-2+\cdots+ck-1*bk+ck$$
$$H\ c2\ldots ck+1 = H\ c1\ldots ck - c1*bk-1\ *b+ck+1\ldots. \quad (1)$$

H is the hash value, c is the character in gram, b is the base number, and k is the number of gram characters.

d. Create sets called windows consisting of i has values. If i = 6, then in one window, there are 6 hash values.

e. Selecting the fingerprint from the hashing results by dividing the hash results based on one w window value, and then selecting the smallest hash value from each of these windows.

Steps b - e, related to requirements Position Independence. By forming a fingerprint from a text document, it can be compared with the fingerprint of other documents regardless of the position of the fingerprint.

2.1.2. Data requirements. Data used to build a plagiarism detection system is a proposal proposal data. The final task management system will be updated by the administration and users as users when uploading proposals. However, users can only upload proposals that have been approved by the supervisor. If the proposal has been approved, it can be stored in a database and used as a reference by other users. The next verification is used to avoid damaging the data in the database. The database used is a final task management database called Sista. The final assignment database will be given an additional table to save the proposal data, which will be named the proposal table.

2.2. System design
The system design phase will be built using the Unified Modelling Language (UML) that supports the concept of Object-Oriented Programming (OOP) based programming models as will be applied at the stage of writing program code. This stage will produce modelling documentation; namely Business Processes, Use Case Diagrams, Use Case Scenarios, Sequence Diagrams, Activity Diagrams [10]**.**

2.3. Implementation
The implementation phase is the process of converting system designs into program codes. The running system has been developed using the language program Page Hypertext Pre-Processor (PHP) which has applied the concept of Object Oriented Programming (OOP). This system also uses local servers and databases to store data that is needed at any time and can be accessed again. Local servers use the XAMPP application that supports Apache to build applications based on the web and database used by MySql (PhpMyadmin).

2.4. Testing
After the implementation process, the next step is testing the system. This research carried out two system testing methods; namely white box testing and testing black box. White Box Testing is a test on the program coding module to ensure that the program code is clear of syntactic or logical errors. Black Box Testing is a test that emphasizes testing system functionality to get the expected results.

3. Results and discussion

3.1. System design
Proposal management system is currently running but only limited to the management of the implementation of the proposal. The existing system will be added a function for checking the level of plagiarism of user proposals. Users will upload files on the system and then the system will check the proposal file. The detection process is done by comparing files with several files stored in a database file.

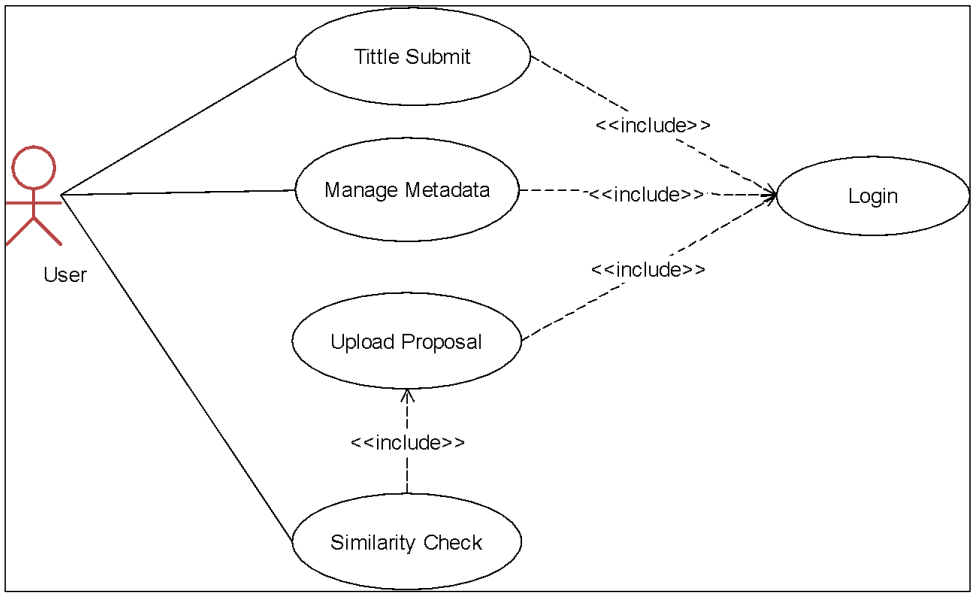

Figure 2. Use case.

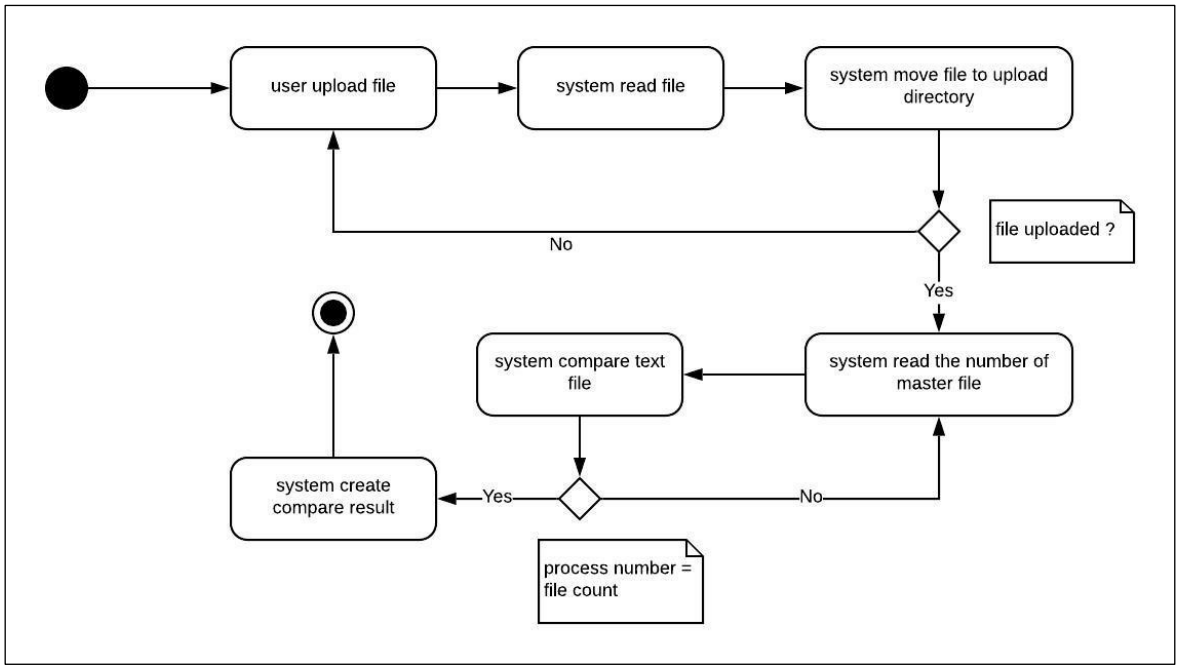

Figure 3. Activity diagram.

The file plagiarism detection process is carried out using a winnowing algorithm. The application of the algorithm Winnowing uses input in the form of a document with pdf type which is then converted into text. The text will then be processed to produce output fingerprint. The fingerprint results of the document will then be compared with the fingerprints of other documents. The test scenarios will be conducted based on the values of the gram, window, and primes base parameters on the five proposal proposals for the informatics management department at Politeknik Negeri Banjarmasin as listed in Table 1 below.

Table 1. Proposal files.

Num	Filename	Users Name
1	Proposal_1.pdf	Molyadin Noor and Defina Qadrunnada
2	Proposal_2.pdf	Hendry and Maulidya Septiyanti
3	Proposal_3.pdf	Nadya Azizah and Ridha Risliana
4	Proposal_4.pdf	Erpan and Herminawati
5	Proposal_5.pdf	Elsa Afrina and Mutia Mariska

3.2. Implementation

Based on the system design, the next stage is system implementation. The implementation of the system is in the form of a plagiarism detection module developed using the php programming language. In the proposal management system, user must log in before they can do the process, as shown in Figure 4.

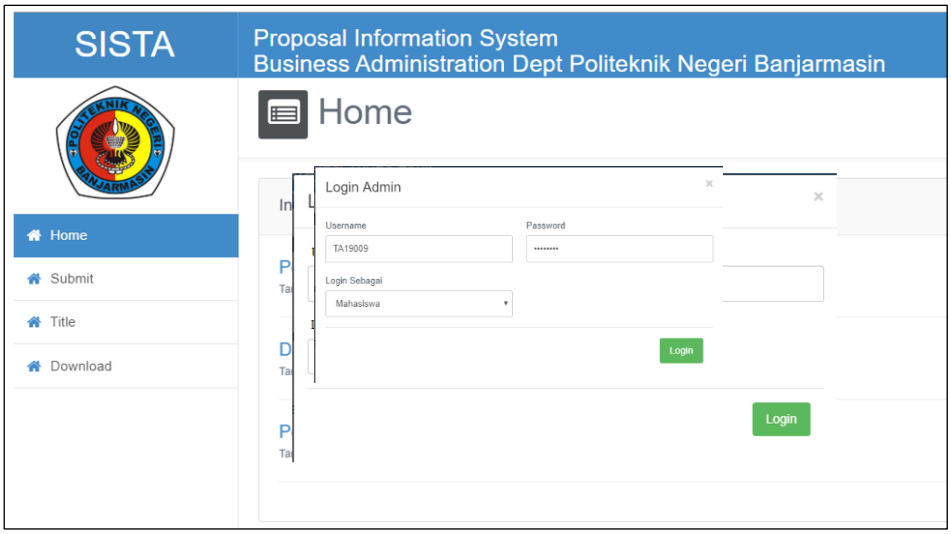

Figure 4. User profile main page.

After the login process is successful, the system will then display the profile page of the user. On this profile page the user can choose to manage the proposal data. To do the plagiarism detection process, the user can choose the Check Proposal menu which will display a page to upload the proposal file in pdf format, as shown in Figure 5.

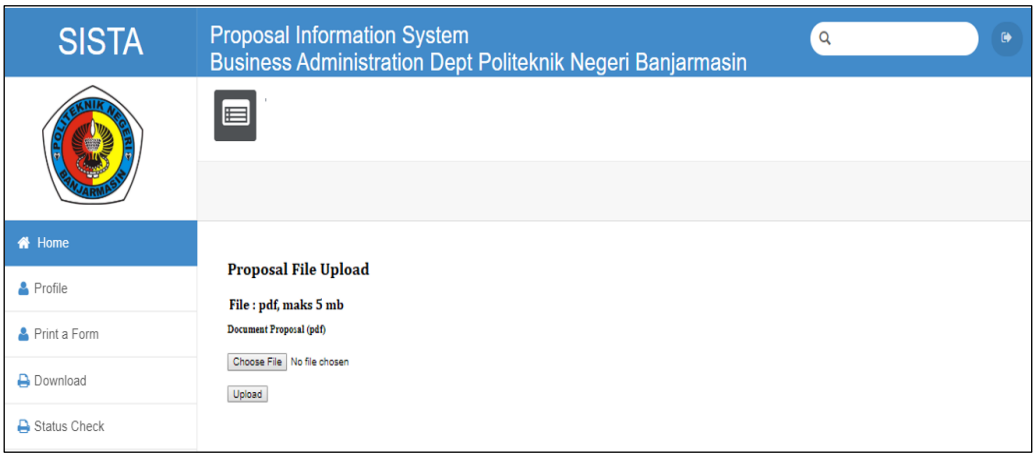

Figure 5. Upload page.

After the pdf extension proposal file is successfully uploaded, the system will proceed to retrieve the text will then be displayed in the text area as shown in Figure 6.

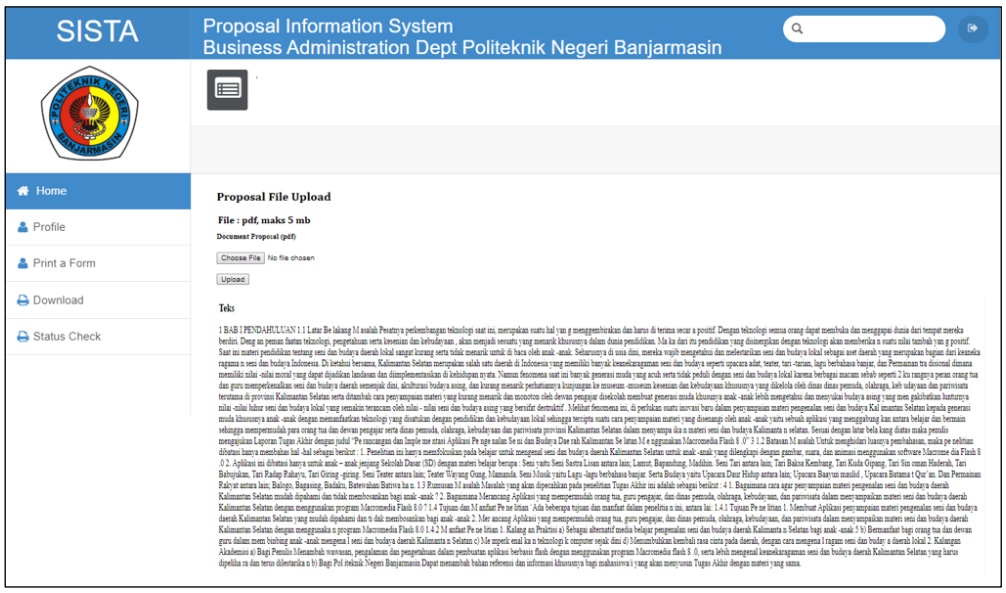

Figure 6. Text display.

On the next page the text of the comparison document is displayed. The comparison document is taken from the location where the comparison pdf file is stored. The system will count the number of files, then repeat the number of documents. An example can be seen in Figure 7, which shows the comparison process of 5 documents carried out 5 times.

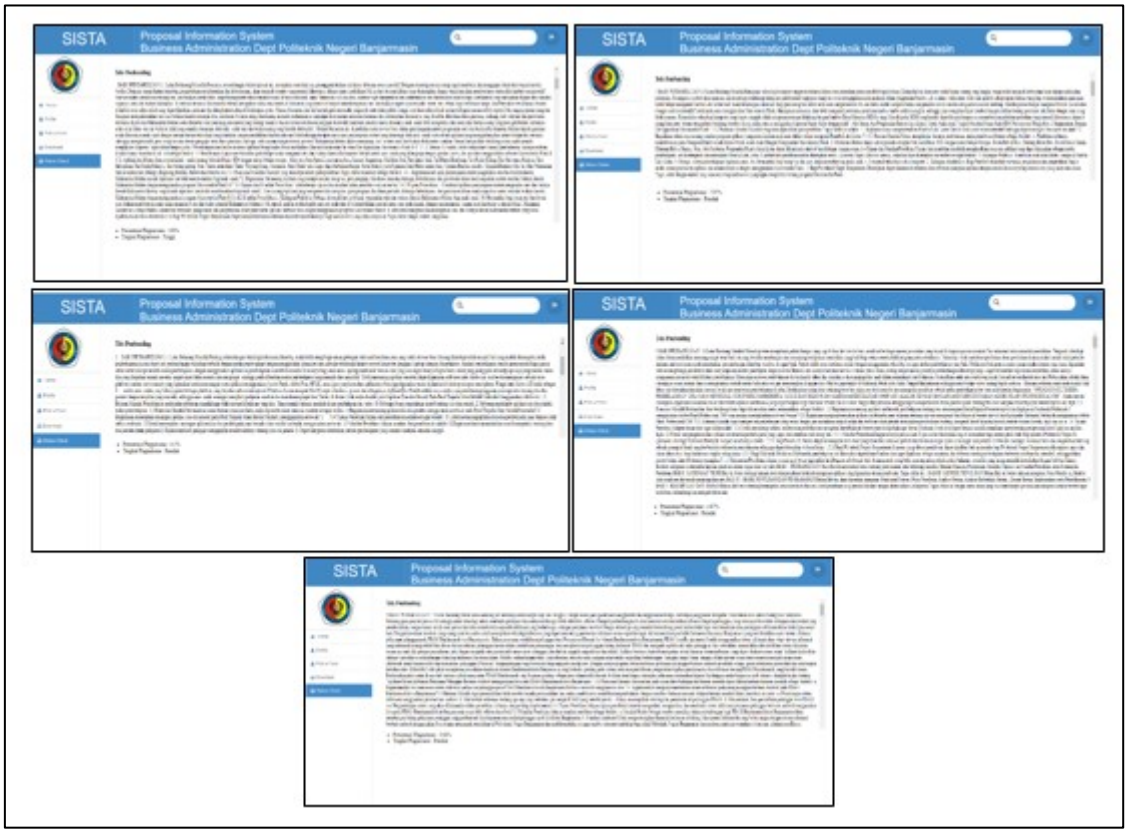

Figure 7. Document comparison process.

3.3. Testing

Tests in this study use 2 types of scenarios. The first scenario is to test the same document but with three different N-gram conditions, window and prime numbers. Because there are 5 documents used for testing, a comparison of 5 times can be done. The test is done by using N Gram = 8, window = 6, prime number = 2. The test results can be seen in Table 2.

Table 2. First scenario 100% testing result.

Document	The Number of Fingerprints Document 1	The Number of Fingerprints Document 2	Union Fingerprints	Intersection	Jaccard Coefficient
Proposal_1.pdf	5062	5062	10124	5062	100%
Proposal_2.pdf	2981	2981	5962	2981	100%
Proposal_3.pdf	2615	2615	5230	2615	100%
Proposal_4.pdf	4905	4905	9810	4905	100%
Proposal_5.pdf	4289	4289	8578	4289	100%

The second scenario for 100% testing using N Gram = 5, prime number = 4, prime number = 2. The test result can be seen in Table 3 below.

Table 3. Second scenario 100% testing result.

Document	The Number of Fingerprints Document 1	The Number of Fingerprints Document 2	Union Fingerprints	Intersection	Jaccard Coefficient
Proposal_1.pdf	5067	5067	10134	5067	100%
Proposal_2.pdf	2986	2986	5972	2986	100%
Proposal_3.pdf	2620	2620	5240	2620	100%
Proposal_4.pdf	4910	4910	9820	4910	100%
Proposal_5.pdf	4294	4294	8588	4294	100%

The third scenario for 100% testing using N Gram = 8, prime number = 6, prime number = 23. The testing result can be seen in Table 4 below.

Table 4. Second scenario 100% testing result.

Document	The Number of Fingerprints Document 1	The Number of Fingerprints Document 2	Union Fingerprints	Intersection	Jaccard Coefficient
Proposal_1.pdf	5062	5062	10124	5062	100%
Proposal_2.pdf	2981	2981	5962	2981	100%
Proposal_3.pdf	2615	2615	5230	2615	100%
Proposal_4.pdf	4905	4905	9810	4905	100%
Proposal_5.pdf	4289	4289	8578	4289	100%

Based on the results of the first scenario test with three different conditions, it can be seen that there is no significant difference. All test results show that the level of similarity of the same document is 100%.

The second scenario is to test different documents with three different N-gram, window and prime number conditions. Because there are 5 documents used for testing, it can be compared 4 times to the first document. The first test is done by using N Gram = 8, window = 6, prime number = 2. The test results can be seen in Table 5.

Table 5. The results of the first document test the first scenario.

Document	The Number of Fingerprints Document 1	The Number of Fingerprints Document 2	Union Fingerprints	Intersection	Jaccard Coefficient
Proposal_2.pdf	5062	2981	8043	1419	21.42%
Proposal_3.pdf	5062	2615	7677	1284	20.08%
Proposal_4.pdf	5062	4905	9967	1983	24.84%
Proposal_5.pdf	5062	4289	9351	1498	19.08%

The second testing for compare different document using N Gram = 5, prime numbers = 4, prime numbers = 2. The test result can be seen in Table 6 below.

Table 6. Results of the first document test second scenario.

Document	The Number of Fingerprints Document 1	The Number of Fingerprints Document 2	Union Fingerprints	Intersection	Jaccard Coefficient
Proposal_2.pdf	5067	2986	8053	3596	80.68%
Proposal_3.pdf	5067	2620	7687	3495	83.37%
Proposal_4.pdf	5067	4910	9977	3758	60.43%
Proposal_5.pdf	5067	4294	9361	3673	64.57%

The third testing for compare different document using N Gram = 5, prime numbers = 4, prime numbers = 2.is performed using N Gram = 8, prime numbers = 6, prime numbers = 23. The test result can be seen in Table 7 below.

Table 7. Test results of the first document third scenario.

Document	The Number of Fingerprints Document 1	The Number of Fingerprints Document 2	Union Fingerprints	Intersection	Jaccard Coefficient
Proposal_2.pdf	5062	2981	8043	585	7.84%
Proposal_3.pdf	5062	2615	7677	283	3.83%
Proposal_4.pdf	5062	4905	9967	429	4.5%
Proposal_5.pdf	5062	4289	9351	326	3.61%

Test results from the second scenario by comparing different documents shows that there are different levels of similarity. However, based on the three tests scenario, it can be seen that the smaller the value of N-Gram, window and prime numbers, the greater the resulting percentage value. From the three conditions, N-Gram = 8, window = 6 and prime number = 23 is the most ideal because they can better represent the real reality.

4. Conclusions

The Winnowing Algorithm is proven to be able to detect similarities between two documents by comparing fingerprints produced from these two documents. Based on testing proves that N-Gram = 8, window = 6 and prime number = 23 is the most effective ones used in detecting plagiarism according to the similarity of each word. The test results prove that the percentage level and execution time are influenced by the number of words. Large n-gram also affects the percentage of similarity, where the similarity percentage of the similarity of words is higher when the n-gram value is getting smaller.

Application of the system is proven to reduce paper usage and computer use. This can be proven by increasing energy efficiency, because to do the document similarity checking process does not require much time and process. In addition, the system can be accessed anywhere using only smartphones, so as to create a reduction in the use of electronic goods. With this system, the parties involved can make changes to lifestyles with a low impact on the environment. When compared with other similarity detection applications that have been tested such as Turnitin, the application developed has advantages including being free and already integrated with the existing system.

5. References

[1] Ercegovac Z and Richardson J V 2004 *College & Research Library* **65**
[2] Ilham I and Pasnur P 2017 *J. Inspiration* **7**

[3] Liu C, Chen C, Han J and Yu P S 2006 *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*
[4] Vikram S 2016 *Proceedings of the 2015 International Conference on Green Computing and Internet of Things* 767–72
[5] Mardiana T, Adji TB and Hidayah I 2015 *Communications in Computer and Information Science*
[6] Ind K S T 2015 An *International Journal of Innovative Research in Computer and Communication Engineering* **03** 3823-3830
[7] Nurdiansyah Y, Muharrom F N and Firdaus 2018 *MATEC Web of Conferences* **164**
[8] Schleimer S, Wilkerson D S, and Aiken A 2003 *ACM SIGMOD International Conference on Management of Data* 76-85
[9] Oetsch J, Pührer J, Schwengerer M, and Tompits H 2010 *Theory and Practice of Logic Programming* **10** 759-775
[10] Page-Jones M 2000 *Fundamentals of Object-Oriented Design in UML* (United States: Addison-Wesley Professional)
[11] Rumbaugh J, Blaha M, Premerlani W, Eddy F and Lorensen WE 1991 *Object-oriented Modeling and Design* (United States: Englewood Cliffs)
[12] Conallen J 2002 *Building Web Applications with UML* (United States: Addison-Wesley Longman Publishing Co., Inc.)