Selangor Science &Technology Review
Vol. 3, No. 1, (2019)

# Plagiarism Detection Application Uses Winnowing Algorithm with Synonym Recognition for Indonesian Text Documents

Riki[1*], Edy[2], Maryanto[3]

[1]Buddhi Dharma University
riki@ubd.ac.id
[2]Buddhi Dharma University
edy@ubd.ac.id
[3]Buddhi Dharma University

**Abstract**: Plagiarism is the act of recognizing someone else's work as a result of personal work without the original owner's permission work. Plagiarism in the form of documents has happened a lot in this digital era. In response to this, through this scientific work a system will be developed that can be used for detect plagiarism between text documents, namely Rejecting Algorithms with Synonym Recognition. Winnowing Algorithm is a document fingerprint method that is used to detect similarities between text documents using hashing techniques. This algorithm was chosen because Reject is one of the best algorithms to get the value of similarity between document text both in terms of accuracy and the performance. From the results of testing in table 8, the conclusion is that the greater the number of grams and window=2 are used, the process time varies and the percentage of similarity varies, tends to decrease. And also by adding the text processing process and the synonym recognition varies greatly increasing the processing time and decreasing the percentage of similarity.

**Keywords**: Plagiarsm, Similarity, Winnowing, Algorithm, Paraphrasing

## 1. Introduction

Plagiarism is the plagiarism or recognition of the work of others by someone who makes the work as his work. People who do plagiarism are called plagiaris or plagiarists. With such restrictions, plagiarism is theft (harsh language, piracy) and plagiaris is a thief (hijacker).

Plagiarism seems to have been entrenched, easy to happen and very felt at the level of S1 education. The inability of control comes from the relatively low quality of the supervisor, the number of counselors that is not proportional to the number of students, the dedication of the university which is still oriented to mere economic benefits, inconsistent attitudes of the relevant university, current library resources, access to information sources difficult to reach, and unclear or even non-existent sanctions against plagiaris. Examples of recent cases are as many as 100 professors at the level of professors, lecturers and lector of the head of the university, doing plagiarism in 2012. During 2012 there were four who were demoted and two were dismissed. In accordance with (Nasional , 2010) concerning Prevention and Control of Plagiarism in Higher Education, the Chancellor must take action against the lecturer who is carrying out the plagiarism. In addition, the Ministry of Education and Culture also found about 400 private universities, committing crimes in the form of falsifying data on the number of lecturers and students. The counterfeiting was carried out to obtain coaching funds and lecturer certification allowances. Realizing this, through this thesis the authors propose a plagiarism detection system to check the level of similarity between documents. (Jody, Wibowo, & Arifianto, 2015)

Clustering documents has long been applied to make it easier for users to search for documents. The application of clustering is standardized on a hypothesis (cluster hypothesis) that the relevant documents will tend to be in the same cluster if clustering is done in the document collection. Several previous studies related to clustering have been carried out, among others: research conducted by mentioning that the effect of stemming has an effect on the accuracy of clustering documents in Arabic. Whereas the IR field research for Indonesian language documents has been carried out by those who examine the effects of stemming on Indonesian. While research in the field of IR which examines the effect of preprocessing text for Indonesian language document clustering is still rarely done. This is because in general the research on language computing for Indonesian documents is still very minimal. Previous research has been conducted by examining the effect of preprocessing text on the detection of plagiarsme. While in this study it was intended to determine the effect of preprocessing text and its combination on the accuracy of Indonesian language text document clustering. (Milatina, Syukur, & Supriyanto, 2012)

There are several algorithms that function to detect plagiarism in documents such as Rabin Karp and Winnowing. The Winnowing algorithm is one of the Document Fingerprinting algorithms that uses hashing techniques to match two or more documents. The hashing technique itself is useful for converting each string in a document to a hash value whose value will be used as the fingerprint of the document. In Winnowing the hashing function used is Rolling Hashing. (Jody, Wibowo, & Arifianto, 2015)

The Winnowing and Synonym Recognition algorithms can only see similarities in characters, so that if a word changes with the same meaning, the system cannot detect the similarity of meaning. Therefore a method is needed to overcome this problem, and the author adds Stemming method to the Nazief-Adriani Algorithm. Stemming is the process of doing search synonyms search for basic words from each word tokenizing results

## 2. Related Works/Literature Review

Plagiarism in the Dictionary of Contemporary Indonesian, Plagiarism is defined as the act of plagiarizing writing, ideas and others belonging to others (KBIK, p. 1172). Whereas in the Indonesian Dictionary, Plagiarism is defined as plagiarism that violates copyright. Thus Plagiarism is an act of abuse, theft/seizure, control, statement, or declare as one's own from a thought, idea, writing, or creation that is actually owned by someone else (Pratama & Cahyono).

In general, a plagiarism detection system is developed for: (Dewanto , Indriati , & Cholissodin)

a. Text data such as essays, articles, journals, research and so on.
b. Text documents are more structured like programming languages

Some types of plagiarism include the following: (Iyer, Parvati , & Abhipsita, 2005)

a. Word-for-word plagiarism This type is included in the act of copying each word directly without changing the sentence structure at all.
b. Plagiarism of authorship The act of recognizing the work of another person as a result of his own work by including his own name and replacing the actual author's name.
c. Plagiarism of ideas Action in the form of recognizing the results of thoughts or ideas of others.
d. Plagiarism of sources if a writer uses quotes from other authors without including the name of the original owner of the source.

Based on the method used, (Meuschke & Gipp, 2013) categorize plagiarism practices as follows:

a. Copy & Paste Plagiarism, copy every word without any changes.

b. Disguised Plagiarism, belonging to the practice of covering the copied part, was identified in four techniques, namely shake & paste, expensive plagiarism, contractive plagiarism, and mosaic plagiarism.
c. Technical Disguise, summarizing techniques to hide plagiarism content from automatic detection by exploiting the weaknesses of the basic text analysis method, for example by replacing letters with foreign letter symbols.
d. Undue Paraphrasing, deliberately rewrote foreign thoughts by choosing plagiarist words and styles by hiding original sources.
e. Translated Plagiarism, converts content from one language to another.
f. Idea Plagiarism, uses foreign ideas without stating original sources.
g. Self Plagiarism, the use of part or all of personal writing that is not scientifically justified.

## 3. Method

### Winnowing Algorithm

The Winnowing Algorithm method is a document fingerprinting algorithm that is used to detect copies of documents using the hashing technique. The Winnowing algorithm is a method that increases the efficiency of the comparison process of fingerprinting documents. In the case of plagiarism detection, this method can identify small similar parts in a large number of documents. The input of this algorithm is a text document that is processed so as to produce output in the form of a collection of hash values. Hash value is a numeric value formed from ASCII calculations for each character. The collection of hash values is then referred to as fingerprint. Fingerprint is used as an indicator to compare similarities between text documents. (Jody, Wibowo, & Arifianto, 2015).

Steps in the winnowing method:

a. removal of irrelevant characters

The text that will be detected is Plagiarism Detection. The first step in implementing the Winnowing algorithm is to lowercase or change each character in the string to lowercase and remove characters from irrelevant documents such as punctuation, spaces and other symbols. Punctuation, spaces and symbols other than the alphabet are said to be irrelevant because their unique values cannot be taken and are not related to the string to be processed so that the following text results are obtained:

"uji file guna deteksi plagiatrisme"

b. formation of gram series with size k

The second step, forming the text of the first step into the k-gram sequence. In this step, the results of the text from the first step in the form of a collection of strings will be grouped into a new set of strings where the new collection of strings is the result of merging the string the first step with the length of the joined string is k. Below is an example of combining strings with length k = 2:

"uji file guna deteksi plagiatrisme"

↓

"ujifile fileguna gunadeteksi deteksiplagiatrisme"

c. calculation of hash values

The third step is the Rolling Hash process to produce a hash value of every gram that is formed. The example in the "ujifile" string and hash base value = 3, using the formula:

$$H_{(c2\ldots ck+1)} = (H_{(c1\ldots ck+1)-}c1 * b^{(k-1)}) * b + c_{(k+1)}$$

Information:
$H_{(c1\ldots c1)}$: hash value
$c_1$: the ASCII value of the $l$ character in the string
$l$: string length
b: hash basis value

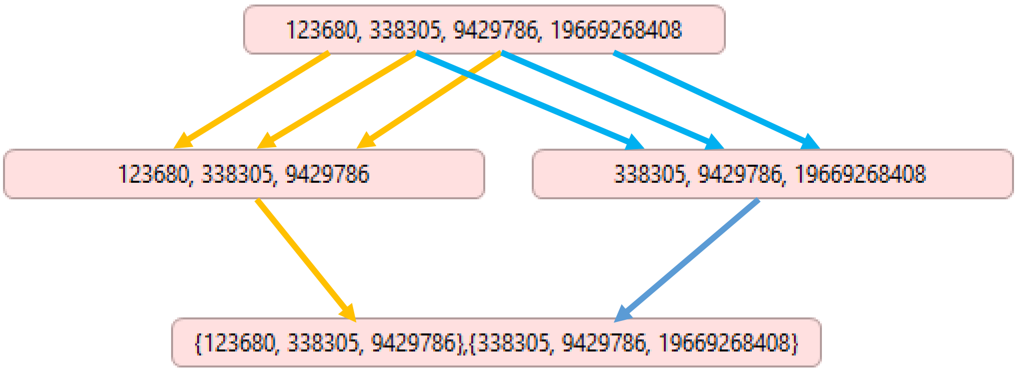
then

$$H_{(c1\ldots c1)} = \left(117x3^{(7-1)}\right) + \left(106x3^{(7-2)}\right)$$
$$+\left(105x3^{(7-3)}\right) + \left(102x3^{(7-4)}\right) + \left(105x3^{(7-5)}\right) + \left(108x3^{(7-6)}\right) + (101)$$
$$= 85293 + 25758 + 8505 + 2754 + 945 + 324 + 101 = 123680$$

From the process, the hash value of each gram is obtained as follows:

$$123680, 338305, 9429786, 19669268408$$

d. divide the hash value into a specific window
After the hash value of each gram is obtained, the fourth step is to form a window. The window formation process is the same as the k-gram process of hash values generated with window = 3:



e. selecting multiple hash values becomes a fingerprinting document
The fifth step is to select the smallest hash value from each window to be used as the fingerprint of the document. The window above shows the following fingerprints:

$$123680, 338305$$

**Synonym Recognition**

Synonym Recognition is the detection of plagiarism through a synonym approach. In this case, document one is compared to other documents by detecting words that contain synonyms. By detecting words that have similar meanings (synonyms) between one document and another can add the value of Similarity so that the results of plagiarism detection are more accurate. Synonym Recognition relies heavily on databases that contain dictionaries synonyms

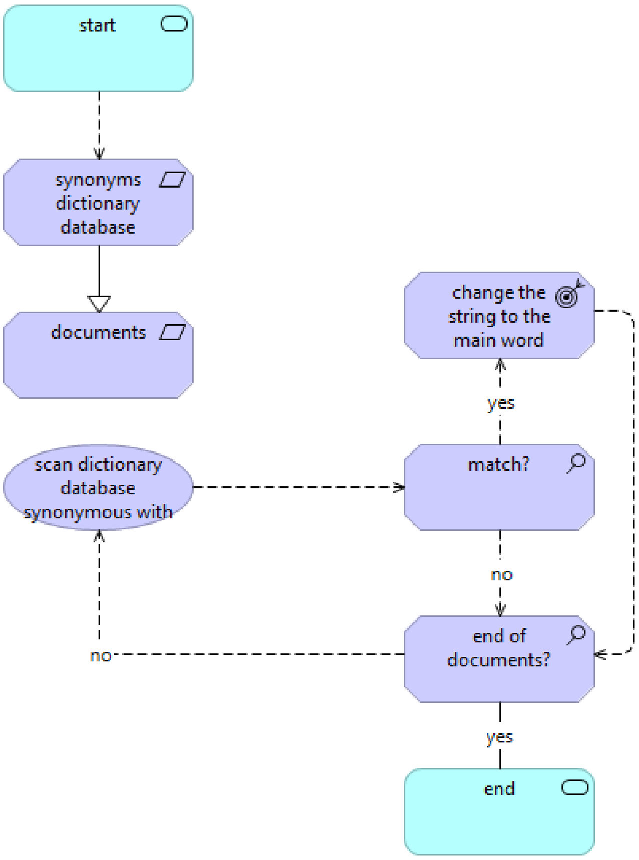

start
synonyms dictionary database
documents
scan dictionary database synonymous with
match?
change the string to the main word
yes
no
end of documents?
no
yes
end

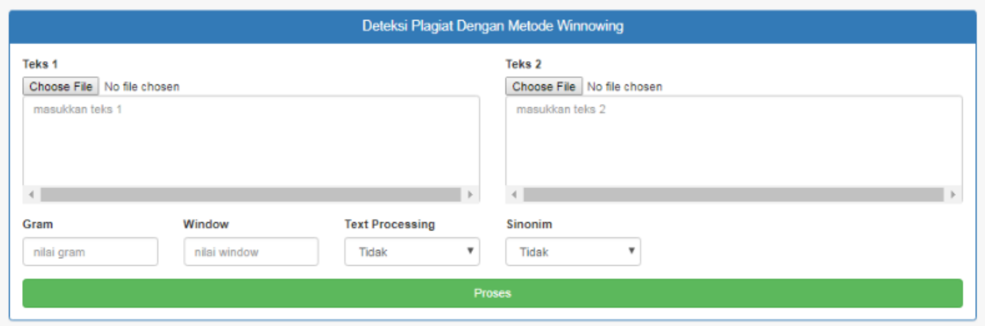
Fig. 1. Synonym Recognition

Document I: "file test for plagiarism detection"
Document II: "file test for plagiarism detection" With Synonym Recognition the word plagiarism will be changed to its main word, cribbing, so that it will become:
Document I: "file test for cribbing detection"
Document II: "file test for cribbing detection"

Winnowing algorithm with Synonym Recognition and adding the Stemming Nazief-Adriani Text Processing feature, because it can improve the accuracy of document similarity detection processes, compared to using the Rabin Karp algorithm tends to be longer even though the level of accuracy is high, based on the test of 7% gram similarity and algorithm Knuth-Morris-Pratt, the similarity of the text produced by the percentage is 41.09%.

In designing this application the author adds a text processing process which includes several stages, namely, folding case, tokenizing, filtering, stemming with the Nazief-Adriani algorithm to improve the accuracy of the similarity of text document detection.

## 4. Results



Deteksi Plagiat Dengan Metode Winnowing

Teks 1: Choose File No file chosen; masukkan teks 1

Teks 2: Choose File No file chosen; masukkan teks 2

Gram: nilai gram

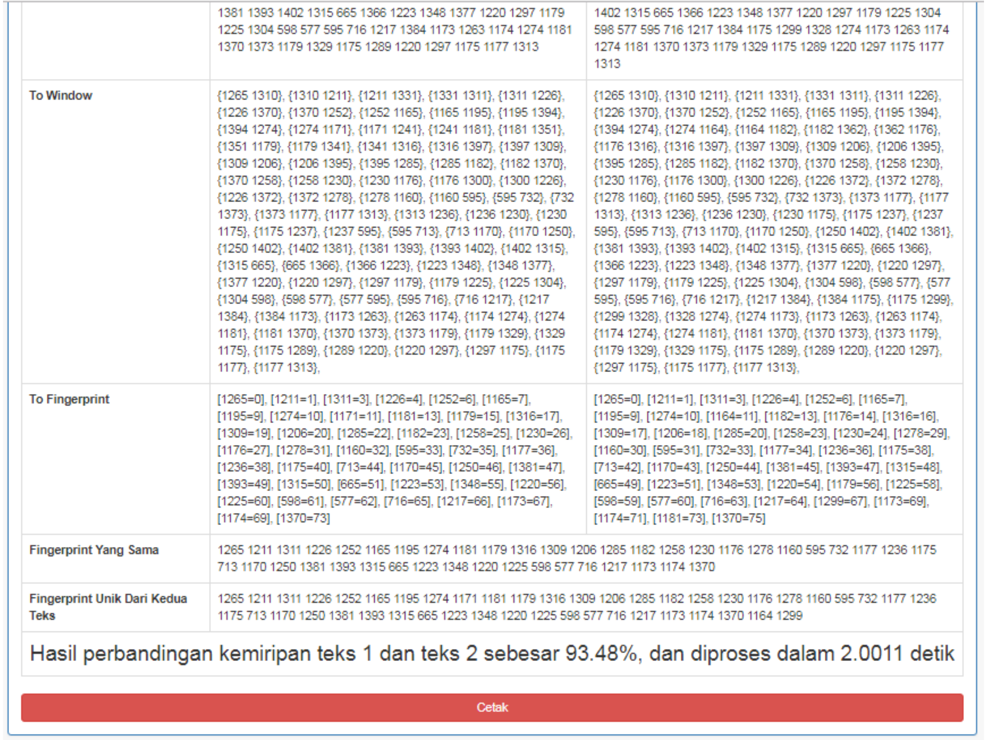
Window: nilai window

Text Processing: Tidak

Sinonim: Tidak

Fig 2. Dashboard

The initial page display of the winnowing method consists of the first text area to input the text to be tested. It can also upload the .doc / .txt file type, the second text area for inputting comparable text can also upload the .doc / .txt file type, combobox gram to select the number of grams, combobox value window to determine how many hash values in one window, combobox text processing and combobox synonyms to select both texts tested through the stages of text processing and synoniom first or not.



| | | |
|---|---|---|
| | 1381 1393 1402 1315 665 1366 1223 1348 1377 1220 1297 1179 1225 1304 598 577 595 716 1217 1384 1173 1263 1174 1274 1181 1370 1373 1179 1329 1175 1289 1220 1297 1175 1177 1313 | 1402 1315 665 1366 1223 1348 1377 1220 1297 1179 1225 1304 598 577 595 716 1217 1384 1175 1299 1328 1274 1173 1263 1174 1274 1181 1370 1373 1179 1329 1175 1289 1220 1297 1175 1177 1313 |
| To Window | {1265 1310}, {1310 1211}, {1211 1331}, {1331 1311}, {1311 1226}, {1226 1370}, {1370 1252}, {1252 1165}, {1165 1195}, {1195 1394}, {1394 1274}, {1274 1171}, {1171 1241}, {1241 1181}, {1181 1351}, {1351 1179}, {1179 1341}, {1341 1316}, {1316 1397}, {1397 1309}, {1309 1206}, {1206 1395}, {1395 1285}, {1285 1182}, {1182 1370}, {1370 1258}, {1258 1230}, {1230 1176}, {1176 1300}, {1300 1226}, {1226 1372}, {1372 1278}, {1278 1160}, {1160 595}, {595 732}, {732 1373}, {1373 1177}, {1177 1313}, {1313 1236}, {1236 1230}, {1230 1175}, {1175 1237}, {1237 595}, {595 713}, {713 1170}, {1170 1250}, {1250 1402}, {1402 1381}, {1381 1393}, {1393 1402}, {1402 1315}, {1315 665}, {665 1366}, {1366 1223}, {1223 1348}, {1348 1377}, {1377 1220}, {1220 1297}, {1297 1179}, {1179 1225}, {1225 1304}, {1304 598}, {598 577}, {577 595}, {595 716}, {716 1217}, {1217 1384}, {1384 1173}, {1173 1263}, {1263 1174}, {1174 1274}, {1274 1181}, {1181 1370}, {1370 1373}, {1373 1179}, {1179 1329}, {1329 1175}, {1175 1289}, {1289 1220}, {1220 1297}, {1297 1175}, {1175 1177}, {1177 1313}, | {1265 1310}, {1310 1211}, {1211 1331}, {1331 1311}, {1311 1226}, {1226 1370}, {1370 1252}, {1252 1165}, {1165 1195}, {1195 1394}, {1394 1274}, {1274 1164}, {1164 1182}, {1182 1362}, {1362 1176}, {1176 1316}, {1316 1397}, {1397 1309}, {1309 1206}, {1206 1395}, {1395 1285}, {1285 1182}, {1182 1370}, {1370 1258}, {1258 1230}, {1230 1176}, {1176 1300}, {1300 1226}, {1226 1372}, {1372 1278}, {1278 1160}, {1160 595}, {595 732}, {732 1373}, {1373 1177}, {1177 1313}, {1313 1236}, {1236 1230}, {1230 1175}, {1175 1237}, {1237 595}, {595 713}, {713 1170}, {1170 1250}, {1250 1402}, {1402 1381}, {1381 1393}, {1393 1402}, {1402 1315}, {1315 665}, {665 1366}, {1366 1223}, {1223 1348}, {1348 1377}, {1377 1220}, {1220 1297}, {1297 1179}, {1179 1225}, {1225 1304}, {1304 598}, {598 577}, {577 595}, {595 716}, {716 1217}, {1217 1384}, {1384 1175}, {1175 1299}, {1299 1328}, {1328 1274}, {1274 1173}, {1173 1263}, {1263 1174}, {1174 1274}, {1274 1181}, {1181 1370}, {1370 1373}, {1373 1179}, {1179 1329}, {1329 1175}, {1175 1289}, {1289 1220}, {1220 1297}, {1297 1175}, {1175 1177}, {1177 1313}, |
| To Fingerprint | [1265=0], [1211=1], [1311=3], [1226=4], [1252=6], [1165=7], [1195=9], [1274=10], [1171=11], [1181=13], [1179=15], [1316=17], [1309=19], [1206=20], [1285=22], [1182=23], [1258=25], [1230=26], [1176=27], [1278=31], [1160=32], [595=33], [732=35], [1177=36], [1236=38], [1175=40], [713=44], [1170=45], [1250=46], [1381=47], [1393=49], [1315=50], [665=51], [1223=53], [1348=55], [1220=56], [1225=60], [598=61], [577=62], [716=65], [1217=66], [1173=67], [1174=69], [1370=73] | [1265=0], [1211=1], [1311=3], [1226=4], [1252=6], [1165=7], [1195=9], [1274=10], [1164=11], [1182=13], [1176=14], [1316=16], [1309=17], [1206=18], [1285=20], [1258=23], [1230=24], [1278=29], [1160=30], [595=31], [732=33], [1177=34], [1236=36], [1175=38], [713=42], [1170=43], [1250=44], [1381=45], [1393=47], [1315=48], [665=49], [1223=51], [1348=53], [1220=54], [1179=56], [1225=58], [598=59], [577=60], [716=63], [1217=64], [1299=67], [1173=69], [1174=71], [1181=73], [1370=75] |
| Fingerprint Yang Sama | 1265 1211 1311 1226 1252 1165 1195 1274 1181 1179 1316 1309 1206 1285 1182 1258 1230 1176 1278 1160 595 732 1177 1236 1175 713 1170 1250 1381 1393 1315 665 1223 1348 1220 1225 598 577 716 1217 1173 1174 1370 | |
| Fingerprint Unik Dari Kedua Teks | 1265 1211 1311 1226 1252 1165 1195 1274 1171 1181 1179 1316 1309 1206 1285 1182 1258 1230 1176 1278 1160 595 732 1177 1236 1175 713 1170 1250 1381 1393 1315 665 1223 1348 1220 1225 598 577 716 1217 1173 1174 1370 1164 1299 | |

Hasil perbandingan kemiripan teks 1 dan teks 2 sebesar 93.48%, dan diproses dalam 2.0011 detik

Fig 3. Process of the Winnowing With Synonym Recognition Method

Displaying the results of the tested text and comparative text, gram = 2, number of windows = 2, choosing to use text processing and using synonyms (YES), displaying detailed calculations so that it can produce document similarities of 93.48% in 2.0011 seconds.

## Testing of the Winnowing Method

Tests carried out on winnowing methods with synonym recogniton namely testing the number of gram values, number of windows, text processing and synonym recognition so as to produce process time and the results of the accuracy of document similarities. The data tested is in the form of 2 plain text pieces.

Text 1 *"Indonesia akhirnya membuka harapan untuk tampil Asian Games ke-18 pada tanggal 18 Agustus - 2 September 2018, didua tempat yaitu Jakarta dan Palembang."*

*Text 2* *"Indonesia akhirnya membuka asa untuk muncul Asian Games ke-18 pada tanggal 18 Agustus - 2 September 2018, didua loka yaitu Jakarta dan Palembang."*

In this test using the number of grams = 2 with the number of windows 2 to 5 and the number of windows = 2 with the number of grams = 2 to 5. To find out the test results can be seen in table 1, table 2, table 3, table 4, table 5, table 6, table 7 and table 8

Table 1. Results of Testing the Winnowing Method with Synonym Recognition with Gram = 2 and the Variation Window Without Text Processing and Synonyms

| Total *Gram* | Total *Window* | Processing Time (without text processing and synonyms) | Percentage of Similarities (without text processing and synonyms) |
|---|---|---|---|
| 2 | 2 | 0.0060 detik | 80% |
| 2 | 3 | 0.0070 detik | 82.22% |
| 2 | 4 | 0.0050 detik | 82.05% |
| 2 | 5 | 0.0060 detik | 75.76% |

From the results of testing in table 1, the conclusion is that gram = 2 the greater the number of windows used, the process time will vary where the percentage of similarity varies tend to decrease as the number of windows changes. And also by adding the text processing process and synonym recognition, however, in table 1 it does not use text processing and synonym recognition

.

Table 2. Results of Testing Winnowing Methods With Synonym Recognition With Gram Variations and Window = 2 Without Text Processing and Synonyms

| Total *Gram* | Total *Window* | Processing Time (without text processing and synonyms) | Percentage of Similarities (without text processing and synonyms) |
|---|---|---|---|
| 2 | 2 | 0.0060 detik | 80% |
| 3 | 2 | 0.0070 detik | 69.05% |
| 4 | 2 | 0.0090 detik | 65.22% |
| 5 | 2 | 0.0080 detik | 61.7% |

From the test results in table 2, the conclusion is that the greater the number of grams and window = 2 is used, the process time will vary where the percentage of similarities varies tend to decrease as the number of windows changes. And also by adding text processing and synonym recognition, however, in table 2 it does not use text processing and synonym recognition.

Table 3. Results of Testing Winnowing Methods with Synonym Recognition with Gram = 2 and Window Variations with Text Processing and without Synonyms

| Total *Gram* | Total *Window* | Processing Time (without text processing and synonyms) | Percentage of Similarities (without text processing and synonyms) |
|---|---|---|---|
| 2 | 2 | 0.8440 detik | 82.98% |
| 2 | 3 | 1.0441 detik | 81.82% |
| 2 | 4 | 1.0531 detik | 86.21% |

| | | | |
|---|---|---|---|
| 2 | 5 | 0.7650 detik | 88% |

From the results of testing in table 3, the conclusion is that gram = 2 and the greater the number of windows used, the processing time varies and the percentage of similarity varies, tends to increase. And also by adding the text processing process and synonym recognition, however, in table 3, not using synonym recognition varies greatly in processing time and increases the percentage of similarities.

Table 4. Results of Testing Winnowing Methods With Synonym Recognition With Gram Variations and Window = 2 With Text Processing and Without Synonyms

| Total *Gram* | Total *Window* | Processing Time (without text processing and synonyms) | Percentage of Similarities (without text processing and synonyms) |
|---|---|---|---|
| 2 | 2 | 0.9191 detik | 82.98% |
| 3 | 2 | 1.0141 detik | 69.84% |
| 4 | 2 | 1.1941 detik | 68.18% |
| 5 | 2 | 0.8580 detik | 62.69% |

From the results of testing in table 4, the conclusion is that the greater the number of grams and window = 2 are used, the process time varies and the percentage of similarity varies, decreasing. And also by adding the text processing process and synonym recognition, however, in table 4 do not use synonym recognition, the results vary in processing time and decreasing percentage of similarity.

Table 5. Results of Testing Winnowing Methods With Synonym Recognition With Gram = 2 and Window Variations Without Text Processing And With Synonyms

| Total *Gram* | Total *Window* | Processing Time (without text processing and synonyms) | Percentage of Similarities (without text processing and synonyms) |
|---|---|---|---|
| 2 | 2 | 0.2260 detik | 100% |
| 2 | 3 | 0.1640 detik | 100% |
| 2 | 4 | 0.1900 detik | 100% |
| 2 | 5 | 0.2830 detik | 100% |

From the results of testing in table 5, the conclusion is that gram = 2 and the greater the number of windows used, the process time varies and the percentage of similarity is 100%. And also by adding text processing and synonym recognition, however, in table 5, not using text processing greatly increases processing time and increases the percentage of similarity by 100%.

Table 6. Results of Testing Winnowing Methods With Synonym Recognition With Gram Variations and Window = 2 Without Text Processing And With Synonyms

| Total *Gram* | Total *Window* | Processing Time (without text processing and synonyms) | Percentage of Similarities (without text processing and synonyms) |
|---|---|---|---|
| 2 | 2 | 0.1200 detik | 100% |
| 3 | 2 | 0.1900 detik | 100% |
| 4 | 2 | 0.1600 detik | 100% |

| | | | |
|---|---|---|---|
| 5 | 2 | 0.2040 detik | 100% |

From the results of testing in table 6, the conclusion is that the greater the number of grams and window = 2 are used, the process time varies and the percentage of similarity is 100%. And also by adding the text processing process and synonym recognition will greatly increase the processing time and increase the percentage of similarity.

Table 7 Results of Testing Winnowing Methods With Synonym Recognition With Gram = 2 and Window Variations With Text Processing And Synonyms

| **Total *Gram*** | **Total *Window*** | **Processing Time (without text processing and synonyms)** | **Percentage of Similarities (without text processing and synonyms)** |
|---|---|---|---|
| 2 | 2 | 1.0181 detik | 93.48% |
| 2 | 3 | 1.4091 detik | 87.88% |
| 2 | 4 | 1.5291 detik | 89.66% |
| 2 | 5 | 1.0081detik | 92% |

From the results of testing in table 7, the conclusion is that gram = 2 the greater the number of windows used, the process time varies and the percentage of similarity varies. And also by adding the text processing process and synonym recognition will increase the process time varies and the percentage of similarity varies too.

Table 8. Results of Testing Winnowing Methods With Synonym Recognition With Gram Variations and Window = 2 With Text Processing And Synonyms

| **Total *Gram*** | **Total *Window*** | **Processing Time (without text processing and synonyms)** | **Percentage of Similarities (without text processing and synonyms)** |
|---|---|---|---|
| 2 | 2 | 1.0181 detik | 93.48% |
| 3 | 2 | 1.1591 detik | 81.36% |
| 4 | 2 | 1.3321 detik | 79.03% |
| 5 | 2 | 1.0881 detik | 73.02% |

From the results of testing in table 8, the conclusion is that the greater the number of grams and window = 2 are used, the process time varies and the percentage of similarity varies, tends to decrease. And also by adding the text processing process and the synonym recognition varies greatly increasing the processing time and decreasing the percentage of similarity.

**Stages and Manual Calculations of the Winnowing With Synonym Recognition Method**
**Step 1:**
In this step text 1 and text 2 are capitalized as a whole and eliminate punctuation.

Text 1 to:
*"Indonesia akhirnya membuka harapan untuk tampil* Asian Games *ke-18 pada tanggal 18 Agustus - 2 September 2018, didua tempat yaitu Jakarta dan Palembang.".*

Text 2 become:
*"Indonesia akhirnya membuka asa untuk muncul* Asian Games *ke-18 pada tanggal 18 Agustus - 2 September 2018, didua loka yaitu Jakarta dan Palembang.".*

**Step 2:**
In the second step, the stoplist process and the stemming process of each text are called text processing.

Text 1 to:
*"indonesia buka harap tampil asi games ke18 tanggal 18 agustus 2 september 2018 dua jakarta Palembang"*

Text 2 become:
*"indonesia buka asa muncul asi games ke18 tanggal 18 agustus 2 september 2018 dua loka jakarta Palembang"*

**Step 3:**
In the third step is the process of synonym recognition of each text.

Text 1 to:
*"indonesia buka harap muncul asi games ke18 tanggal 18 agustus 2 september 2018 dua jakarta Palembang"*

Text 2 become:
*"indonesia buka asa muncul asi games ke18 tanggal 18 agustus 2 september 2018 dua loka jakarta palembang"*

**Step 4:**
In the fourth step, the process of dividing words into grams where in this manual calculation uses gram = 2. So as to produce the following values.

Text 1 to:
*"in **nd do** on ne es si ia ab bu uk ka ah ha ar ra ap pm mu un nc cu ul la as si ig ga am me es sk ke e1 18 8t ta an ng gg ga al l1 18 8a ag gu us st tu us s2 2s se ep pt te em mb be er r2 20 01 18 8d du ua aj ja ak ka ar rt ta ap pa al le em mb ba an ng"*

Text 2 become:
*"in nd do on ne es **si ia** ab bu uk ka aa as sa am mu un nc cu ul la as si ig ga am me es sk ke e1 18 8t ta an ng gg ga al l1 18 8a ag gu us st tu us s2 2s se ep pt te em mb be er r2 20 01 18 8d du ua al lo ok ka aj ja ak ka ar rt ta ap pa al le em mb ba an ng"*

**Step 5:**
In the fifth step, the process of rolling hashes from the results of each gram formed. The following are examples of manual calculations of several gram values:

Calculates the hash value of a word "**nd**":
$= (ascii(o)*11^{1)} + (ascii\ (n)*11^{0})$
$= (110 * 11) + (100* 1)$
$= 1210 + 100$
$= 1310$

Calculates the hash value of a word "**do**":
$= (ascii(n)*11^{1}) + (ascii(o)*11^{0})$
$= (100* 11) + (111 * 1)$
$= 110 + 111$
$= 1211$

Calculates the hash value of a word "**si**" :
$= (ascii(s)*11^{1}) + (ascii(n)*11^{0})$

= (115 * 11) + (105 * 1)
= 1265 + 105
= 1370
Calculates the hash value of a word "**ia**":
= (ascii(n)*$11^1$) + (ascii(i)*$11^0$)
= (105 * 11) + (97 * 1)
= 1155 + 97
= 1252

From the above calculation example and applied to each gram so as to produce the following values

Text 1 to:
1265 1310 1211 1331 1311 1226 1370 1252 1165 1195 1394 1274 1171 1241 1181 1351 1179 1341 1316 1397 1309 1206 1395 1285 1182 1370 1258 1230 1176 1300 1226 1372 1278 1160 595 732 1373 1177 1313 1236 1230 1175 1237 595 713 1170 1250 1402 1381 1393 1402 1315 665 1366 1223 1348 1377 1220 1297 1179 1225 1304 598 577 595 716 1217 1384 1173 1263 1174 1274 1181 1370 1373 1179 1329 1175 1289 1220 1297 1175 1177 1313

Text 2 become:
1265 1310 1211 1331 1311 1226 1370 1252 1165 1195 1394 1274 1164 1182 1362 1176 1316 1397 1309 1206 1395 1285 1182 1370 1258 1230 1176 1300 1226 1372 1278 1160 595 732 1373 1177 1313 1236 1230 1175 1237 595 713 1170 1250 1402 1381 1393 1402 1315 665 1366 1223 1348 1377 1220 1297 1179 1225 1304 598 577 595 716 1217 1384 1175 1299 1328 1274 1173 1263 1174 1274 1181 1370 1373 1179 1329 1175 1289 1220 1297 1175 1177 1313

**Step 6:**
The sixth step is to form a window of each formed hash value. Where window = 3. Example of forming a window of several hash values, namely:

[1265, 1310, 1211, 1331, 1311]
become
{1265 1310 **1211**}, {1310 **1211** 1331}, {**1211** 1331 1311}

From the example of forming the window above and applying it to each gram value it will produce the following values:
Text 1 to:
{1265 1310 **1211**}, {1310 **1211** 1331}, {**1211** 1331 1311}, {1331 1311 **1226**}, {1311 **1226** 1370}, {**1226** 1370 1252}, {1370 1252 **1165**}, {1252 **1165** 1195}, {**1165** 1195 1394}, {**1195** 1394 1274}, {1394 1274 **1171**}, {1274 **1171** 1241}, {**1171** 1241 1181}, {1241 **1181** 1351}, {1181 1351 **1179**}, {1351 **1179** 1341}, {**1179** 1341 1316}, {1341 **1316** 1397}, {1316 1397 **1309**}, {1397 1309 **1206**}, {1309 **1206** 1395}, {**1206** 1395 1285}, {1395 1285 **1182**}, {1285 **1182** 1370}, {**1182** 1370 1258}, {1370 1258 **1230**}, {1258 1230 **1176**}, {1230 **1176** 1300}, {**1176** 1300 1226}, {1300 **1226** 1372}, {**1226** 1372 1278}, {1372 1278 **1160**}, {1278 1160 **595**}, {1160 **595** 732}, {**595** 732 1373}, {**732** 1373 1177}, {1373 **1177** 1313}, {**1177** 1313 1236}, {1313 1236 **1230**}, {1236 1230 **1175**}, {1230 **1175** 1237}, {**1175** 1237 595}, {1237 **595** 713}, {**595** 713 1170}, {**713** 1170 1250}, {**1170** 1250 1402}, {**1250** 1402 1381}, {1402 **1381** 1393}, {**1381** 1393 1402}, {1393 1402 **1315**}, {1402 1315 **665**}, {1315 **665** 1366}, {**665**

1366 1223}, {1366 **1223** 1348}, {**1223** 1348 1377}, {1348 1377 **1220**}, {1377 **1220** 1297}, {1220 1297 **1179**}, {1297 **1179** 1225}, {**1179** 1225 1304}, {1225 1304 **598**}, {1304 598 **577**}, {598 **577** 595}, {577 **595** 716}, {**595** 716 1217}, {**716** 1217 1384}, {1217 1384 **1173**}, {1384 **1173** 1263}, {**1173** 1263 1174}, {1263 **1174** 1274}, {**1174** 1274 1181}, {1274 **1181** 1370}, {**1181** 1370 1373}, {1370 1373 **1179**}, {1373 **1179** 1329}, {1179 1329 **1175**}, {1329 **1175** 1289}, {**1175** 1289 1220}, {1289 1220 1297}, {1220 1297 **1175**}, {1297 **1175** 1177}, {**1175** 1177 1313}

Text 2 become:
{1265 1310 **1211**}, {1310 **1211** 1331}, {**1211** 1331 1311}, {1331 1311 **1226**}, {1311 **1226** 1370}, {**1226** 1370 1252}, {1370 1252 **1165**}, {1252 **1165** 1195}, {**1165** 1195 1394}, {**1195** 1394 1274}, {1394 1274 **1164**}, {1274 **1164** 1182}, {**1164** 1182 1362}, {1182 1362 **1176**}, {1362 **1176** 1316}, {**1176** 1316 1397}, {1316 1397 **1309**}, {1397 1309 **1206**}, {1309 **1206** 1395}, {**1206** 1395 1285}, {1395 1285 **1182**}, {1285 **1182** 1370}, {**1182** 1370 1258}, {1370 1258 **1230**}, {1258 1230 **1176**}, {1230 **1176** 1300}, {**1176** 1300 1226}, {1300 **1226** 1372}, {**1226** 1372 1278}, {1372 1278 **1160**}, {1278 1160 **595**}, {1160 **595** 732}, {**595** 732 1373}, {**732** 1373 1177}, {1373 **1177** 1313}, {**1177** 1313 1236}, {1313 1236 **1230**}, {1236 1230 **1175**}, {1230 **1175** 1237}, {1175 1237 **595**}, {1237 **595** 713}, {**595** 713 1170}, {**713** 1170 1250}, {**1170** 1250 1402}, {**1250** 1402 1381}, {1402 **1381** 1393}, {**1381** 1393 1402}, {1393 1402 **1315**}, {1402 1315 **665**}, {1315 **665** 1366}, {**665** 1366 1223}, {1366 **1223** 1348}, {**1223** 1348 1377}, {1348 1377 **1220**}, {1377 **1220** 1297}, {1220 1297 **1179**}, {1297 **1179** 1225}, {**1179** 1225 1304}, {1225 1304 **598**}, {1304 598 **577**}, {598 **577** 595}, {**577** 595 716}, {**595** 716 1217}, {**716** 1217 1384}, {1217 1384 **1175**}, {1384 **1175** 1299}, {**1175** 1299 1328}, {1299 1328 **1274**}, {1328 1274 **1173**}, {1274 **1173** 1263}, {**1173** 1263 1174}, {1263 **1174** 1274}, {**1174** 1274 1181}, {1274 **1181** 1370}, {**1181** 1370 1373}, {1370 1373 **1179**}, {1373 **1179** 1329}, {1179 1329 **1175**}, {1329 **1175** 1289}, {**1175** 1289 1220}, {1289 **1220** 1297}, {1220 1297 **1175**}, {1297 **1175** 1177}, {**1175** 1177 1313}

**Step 7:**
The eighth step is to determine the fingerprint document. Fingerprint documents are taken from the smallest value of each window and if from the second value the window is the same, the first one will be taken (so there is no same value).
Next is the fragment of the process of looking for the smallest value in each window where the fingerprint is formed:

**{1265 1310** 1211**}, {1331 1311** 1226**}, {1370 1252** 1165**}**
become
**[**1211**=0], [**1226**=3], [**1165**=6]**

Text 1 to:
[1211=0], [1226=3], [1165=6], [1195=9], [1171=10], [1181=13], [1179=14], [1316=17], [1309=18], [1206=19], [1182=22], [1230=25], [1176=26], [1160=31], [595=32], [732=35], [1177=36], [1175=39], [713=44], [1170=45], [1250=46], [1381=47], [1315=49], [665=50], [1223=53], [1220=55], [598=60], [577=61], [716=65], [1173=66], [1174=69]

Text 2 become:
[1211=0], [1226=3], [1165=6], [1195=9], [1164=10], [1176=13], [1309=16], [1206=17], [1182=20], [1230=23], [1160=29], [595=30], [732=33], [1177=34], [1175=37], [713=42], [1170=43], [1250=44], [1381=45], [1315=47], [665=48], [1223=51], [1220=53], [1179=55], [598=58], [577=59], [716=63], [1274=67], [1173=68], [1174=71], [1181=73]

**Step 8:**
In the ninth step is to combine the fingerprint value of the document from the two pieces of text and look for the same value between the two pieces of text. And after that, the percentage of document similarity is calculated using jaccard's similarity coefficient:

**| A ∩ B |**:
1211 1226 1165 1195 1181 1179 1309 1206 1182 1230 1176 1160 595 732 1177 1175 713 1170 1250 1381 1315 665 1223 1220 598 577 716 1173 1174

**| A ∪ B |** :
1211 1226 1165 1195 1171 1181 1179 1316 1309 1206 1182 1230 1176 1160 595 732 1177 1175 713 1170 1250 1381 1315 665 1223 1220 598 577 716 1173 1174 1164 1274
**D (A, B)** = (29/33) * 100% = **87.88%**

## 5. Conclusion

The conclusions obtained in this study include the following:

a. The winnowing method with synonym recognition can be implemented into an application to detect plagiarism in Indonesian text documents.
b. Results of the method.
c. From the results of testing in table 7, the conclusion is that gram = 2 the greater the number of windows used, the process time varies and the percentage of similarity varies. And also by adding the text processing process and synonym recognition will increase the process time varies and the percentage of similarity varies too
d. The calculation process of the percentage of similarity of text documents is very dependent on the parameters entered in the form of grams, windows, text processing and synonym recogniton

## 6. References

Dewanto , S., Indriati , & Cholissodin, I. (n.d.). *Deteksi Plagiarisme Dokumen Teks Menggunakan Algoritma Rabin-Karp dengan Synonym Recognition.* Malang: Universitas Brawijaya.

Iyer, Parvati , & Abhipsita. (2005). Document Silimarity Analysis for a Plagiarism Detection System. *2nd Indian International Conference on Artificial Intelegence (IICAI-05)*, (pp. 2534-2544).

Jody, Wibowo, A. T., & Arifianto, A. (2015). Analsis dan Implementasi Algoritma Winnowing dengan Synonym Recognition pada Deteksi Plagiarisme untuk Dokumen Teks Berbahasa Indonesia. *e-Proceeding of Engineering* (p. 7674). Bandung: Universitas Telkom.

Meuschke, N., & Gipp, B. (2013). State-of-the-art in detecting academic plagiarism. *International Journal for Educational Integrity*, 50-71.

Milatina, Syukur, A., & Supriyanto, C. (2012). Pengaruh Text Prepocessing Pada Clustering Dokumen Teks Berbahasa Indonesia. *Jurnal Teknologi Informasi*, 29-39.

Nasional , M. P. (2010, Agustus 16). Pencegahan dan Penanggulangan PLagiat. Jakarta, DKI Jakarta, Indonesia.

Pratama , R., & Cahyono, B. (n.d.). *Aplikasi Pendeteksi Duplikasi Dokumen Teks Bahasa Indonesia Menggunakan Algoritma Winnowig Dengan Metode K-Gram Dan Synonym Recognition.* Malang: Universitas Muhammadiyah Malang.