3rd International Conference on Information Science and Technology Innovation (ICoSTEC)
July 27, 2024, Yogyakarta, Indonesia

Implementation of a Plagiarism Detection System Text Based

Progresif Buulolo[1], B. Herawan Hayadi[2], Dedi Hartama[3]

[1,2,3] *Magister of Computer Science, Potensi Utama University*

JL. KL. Yos Sudarso Km. 6.5 No. 3-A, Medan Indonesia

[1]gracebuulolo@gmail.com

[2]b.herawan.hayadi@gmail.com

[3]dedyhartama@amiktunasbangsa.ac.id

Abstract — **Plagiarism, the act of plagiarizing or stealing work without acknowledgment, is a serious challenge in the academic world. Scientific work, as a common target for plagiarism, is increasingly influenced by information technology. This research implements a text-based plagiarism detection system by comparing the level of similarity between the Cosine Similarity and Jaccard Similarity algorithms against winnowing for text similarity detection related to variations in N-gram values 3, 5 and 7. Testing was carried out using the Python programming language and its supporting libraries on 20 datasets sentence. The test results show that Cosine Similarity is better at detecting similarities between texts. Accuracy analysis using the confusion matrix produces an accuracy value of 50%. The comparison results of different n-gram variations have a total performance similarity of 15.89% and an average of 0.26%. Meanwhile, the total performance of Jaccard similarity is 13.59% and the average is 0.23%. Although Cosine Similarity has higher accuracy than Jaccard Similarity, the stability does not reach 100%.**

Keywords — **Similarity Plagiarism, Cosine Similarity, Jaccard Similarity, N-Gram, Text Data**

I. INTRODUCTION

Plagiarism is defined as "to include both theft or misappropriation of intellectual property and the substantial unattributed textual copying of another's work". Intellectual products containing ideas, data and writing are forms of work that often become the subject of plagiarism. Plagiarism can occur intentionally or unintentionally, these two motivational reasons are still considered plagiarism if there are similarities in two scientific works without citing and changing the original text using one's own words [1] . Scientific work is a piece of writing or essay that is obtained in accordance with its scientific nature and is based on various results of observations, research and reviews of certain fields of science. Not a few people make scientific works and it takes a lot of time, most people do it by copying (copying). -paste) other people's work [2] . Plagiarism comes from the English plagiary and the Latin plagiarius which means plagiarist or thief. So plagiarism has the meaning of plagiarizing someone's idea or work which will be recognized as one's own work without stating the source where the idea or idea came from [3] . from the results of other people's thoughts and make it look like their own theory [4] . One example of technology that helps human activities is an information system and the negative is that people tend to do things that are instantaneous, in the form of plagiarism [5] . Winnowing is an algorithm used in the document fingerprinting process by utilizing the hashing function of the text. After that, all the selected hash values are then saved as a fingerprint for the document. This fingerprint will later be used as a basis for comparing similarities between the texts that have been entered [6] . Cosine similarity is used to calculate similarity values by equating words for words and is one of the techniques for measuring text similarity with a high level of accuracy by [7] . By [8] analyzing the highest similarity with the string similarity cosine similarity preprocessing combination algorithm, the experimental results show that its use produces the highest distribution values and similarity ratios. Another research [9] uses the cosine similarity method in normalizing the length of data vectors by comparing N-grams that are parallel to each other from 2 comparisons which aims to create a web-based system that can classify documents automatically using the cosine similarity algorithm in the clustering process using weighting TF-IDF, but the results of the discussion do not provide TF-IDF weighting classification results.

Jaccard Similarity is an algorithm that functions to compare two documents by calculating the similarities or differences of the documents. To run n-gram and Jaccard similarity, an algorithm is needed that functions as a document fingerprint and the algorithm that will be used to support this is the winnowing algorithm [10] . Another study [11] used the N-gram winnowing algorithm method with the Jaccard Similarity method. N-grams have a high influence on the similarity results for taking n pieces of letters.

Several previous studies are references for this research, including research with the research title "Implementation of the Winnowing Algorithm in Detecting Plagiarism in Title and Abstract of Student's Final Project". The application of the winnowing method can help in finding similar results from the title and abstract of students' final assignments with the run test scenario, namely n = 7 with a percentage of 3.07% related after testing [12] . The next research entitled "Tetun Language Plagiarism Detection With Text Mining Approach Using N-gram and Jaccard Similarity Coefficient", this research extracts text documents, then uses the n-gram method to take letters from several strings of a word to end the document with 4-grams. The results of this research are able to show the

performance of the plagiarism detection system for all categories with a similarity percentage above 98.99% [13] . by [5] This research only focuses on the application for detecting plagiarism in text document files and does not pay attention to reference sources, the size of the dataset used only tests 10 text documents, the size of the dataset used is limited so it does not cover wide enough variations to generalize the results. Low accuracy performance, the test results show that the largest level of accuracy produced is 47.58%, while the smallest level of accuracy is only 19.28% and does not provide more detailed information about the level of plagiarism such as the level of similarity between the original source and the document being tested.

Several studies have made plagiarism detection using the same functional algorithm as document fingerprinting, but the problems that often occur in the results of k-grams, n-grams and the level of accuracy based on gram values are different. Based on this background, the author is interested in conducting a good performance analysis of the results of the comparison of the Cosine Similarity and Jaccard Similarity algorithms in text detection against the winnowing algorithm to find the level of similarity between two texts.

II. Research Methods

In this research, designing the system starts from several stages, namely: conducting literature studies, data collection, data preprocessing, design analysis for the implementation of the cosine similarity and Jaccard similarity models, data testing and visualization of the results. The following is the process flow of the research methodology stages in Fig.1.

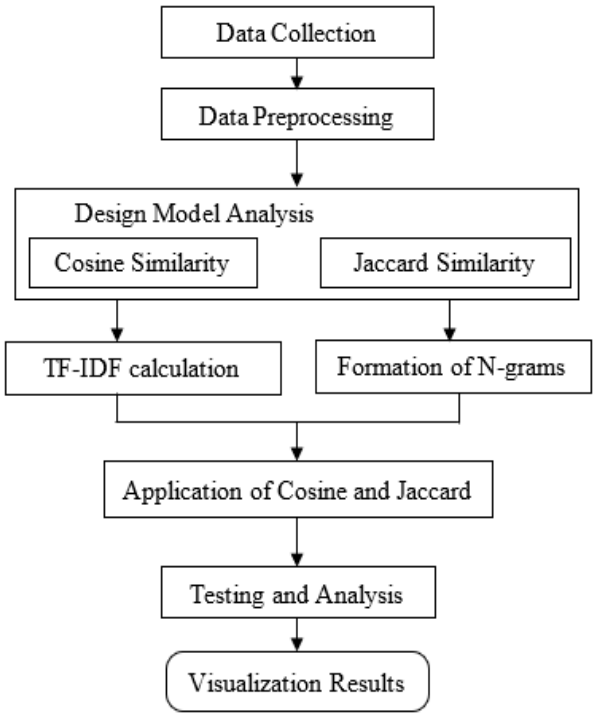

Fig. 1 Stages of Research Methodology

A. Data Collection

This data collection stage was carried out to obtain the information needed to achieve the objectives of this research. In this research, the data is in the form of text, sourced from primary data created by the author in conducting research on the implementation of the two algorithms to calculate the similarity between two texts. The following datasets are used to check the similarity between texts, twenty (20) datasets:

TABLE 1
Example of Dataset Source Data and Target Data

No.	Original Text (Source)	Test Text (Target)
1.	Penelitian Ini Bertujuan Untuk Menginvestigasi Fenomena Dalam Dunia Akademik	Penelitian Ini Bertujuan Untuk Mengekplorasi Fenomena Yang Terus Meningkat
2.	Pendidikan Merupakan Kunci Sukses	Belajar Adalah Fondasi Kesuksesan
3.	Perubahan Iklim Mempengaruhi Kehidupan Sehari-Hari	Dampak Perubahan Iklim Pada Kehidupan Sehari-Hari
4.	Teknologi Informasi Telah Merevolusi Dunia	Revolusi Teknologi Informasi
5.	Kesehatan Mental Penting Untuk Kesejahteraan	Kesehatan Mental Dan Kesejahteraan
6.	Globalisasi Telah Meningkatkan Konektivitas Dunia	Dampak Positif Dan Negatif Dari Globalisasi
7.	Inovasi Menjadi Kunci Keberhasilan Perusahaan	Strategi Inovasi Dalam Dunia Bisnis
8.	Pangan Organik Lebih Sehat Daripada Pangan Konvensional	Mitos Dan Fakta Tentang Makanan Organik
9.	Pariwisata Berkontribusi Pada Pertumbuhan Ekonomi	Manfaat Ekonomi Dari Industri Pariwisata
10.	Pembelajaran Online Semakin Populer	Tantangan Dan Manfaat Pembelajaran Online
11.	Sosial Media Dan Dampaknya Pada Masyarakat	Analisis Dampak Sosial Media
12.	Energi Terbarukan Sebagai Solusi Masa Depan	Pilihan Energi Untuk Keberlanjutan
13.	Keterampilan Interpersonal Dalam Dunia Kerja	Mengembangkan Keterampilan Interpersonal
14.	Hutan Hujan Tropis Dan Perlindungan Lingkungan	Pentingnya Melestarikan Hutan Hujan Tropis

15.	Robotika Dan Otomatisasi Dalam Industri	Transformasi Industri Melalui Robotika
16.	Kebebasan Berpendapat Dan Batasannya	Tantangan Kebebasan Berpendapat Di Era Digital
17.	Gaya Hidup Berkelanjutan Untuk Masa Depan	Langkah-Langkah Menuju Gaya Hidup Berkelanjutan
18.	Pemberdayaan Perempuan Dalam Dunia Kerja	Mendorong Kesataran Gender Di Tampat Kerja
19.	Keamanan Cyber Manjadi Prioritas Utama	Strategi Keamanan Cyber Untuk Organisasi
20.	Manfaat Olahraga Untuk Kesehatan Mental	Manfaat Olahraga Untuk Kesehatan Mental

B. Preprocessing Training Data

This pre-processing stage is the initial stage in data processing to produce a clean dataset before proceeding to a further stage, which consists of case folding, removing spaces between characters and removing special characters.

C. Case Folding

Researchers made all characters uniform, eliminating irrelevant characters in text documents, such as punctuation, spaces and changing uppercase letters to lowercase. This is done so that the two same words do not contain different meanings, for example "This research aims to investigate phenomena in the academic world" and "this research aims to investigate phenomena in the academic world" have the same meaning but are different for the computer. The following is a table of results from the dataset after the case folding process is carried out.

TABLE 2
EXAMPLE OF CASE FOLDING

No.	Original Text (Source)	Test Text (Target)
1.	penelitian ini bertujuan untuk menginvestigasi fenomena dalam dunia akademik	penelitian ini bertujuan untuk mengekplorasi fenomena yang terus meningkat
2.	pendidikan merupakan kunci sukses	belajar adalah fondasi kesuksesan
3.	perubahan iklim mempengaruhi kehidupan sehari-hari	dampak perubahan iklim pada kehidupan sehari hari
4.	teknologi informasi telah merevolusi dunia	revolusi teknologi informasi
5….	kesehatan mental penting untuk kesejahteraan	kesehatan mental dan kesejahteraan
20.	manfaat olahraga untuk kesehatan mental	manfaat olahraga untuk kesehatan mental

D. Remove Spaces Between Characters

At this stage, the process of removing spaces between characters and removing irrelevant punctuation characters and special characters such as !,@,#,$,%,&,^,*, can make the word processing process more efficient in building models and improve determination performance. text similarity. The following are the results of characters that have had spaces removed from the text.

TABLE 3
STOP WORD SPACE BETWEEN CHARACTERS

Content Text	
Original Text (Source)	**Test Text (Target)**
penelitianinibertujuanntukmenginvestigasifenomenadalamduniaakademik	penelitianinibertujuanuntukmengekplorasifenomenayangterusmeningkat
pendidikanmerupakankuncisukses	belajaradalahfondasikesuksesan
perubahaniklimmempengaruhikehidupanseharihari	dampakperubahaniklimpadakehidupanseharihari
teknologiinformasitelahmerevolusidunia	revolusiteknologiinformasi
kesehatanmentalpentinguntukkesejahteraan	kesehatanmentaldankesejahteraan
manfaat olahraga untuk kesehatan mental	manfaat olahraga untuk kesehatan mental

E. Formation of N-Grams

After the text document goes through the pre-processing stage. Then proceed to the n-gram formation stage in the Jaccard algorithm for winnowing, namely the process of solving text strings which are grouped based on the n-gram value which will be calculated continuously moving forward a number of n until the end of the text, the result of forming an n-gram with a value of N=3, 5 and 7. At the stage of the process carried out to cut sentences into parts or words. The part that is broken down is called a token. Following are the results of tokenization of two datasets from a dataset of (20) different n-gram values:

TABLE 4
EXAMPLE OF TOKENIZING N-GRAM DATASET

No.	Stop Word text	N-Gram with N=3
1.	penelitian inibertuju anuntuk menginve stigasifeno menadalam dunia akademik	['pen', 'ene', 'nel', 'eli', 'lit', 'iti', 'tia', 'ian', 'ani', 'nin', 'ini', 'nib', 'ibe', 'ber', 'ert', 'rtu', 'tuj', 'uju', 'jua', 'uan', 'anu', 'nun', 'unt', 'ntu', 'tuk', 'ukm', 'kme', 'men', 'eng', 'nge', 'gek', 'ekp', 'kpl', 'plo', 'lor', 'ora', 'ras', 'asi', 'sif', 'ife', 'fen', 'eno', 'nom', 'ome', 'men', 'ena', 'nay',

		'aya', 'yan', 'ang', 'ngt', 'gte', 'ter', 'eru', 'rus', 'usm', 'sme', 'men', 'eni', 'nin', 'ing', 'ngk', 'gka', 'kat']
		N-Gram dengan N=5
		['penel', 'eneli', 'nelit', 'eliti', 'litia', 'itian', 'tiani', 'ianin', 'anini', 'ninib', 'inibe', 'niber', 'ibert', 'bertu', 'ertuj', 'rtuju', 'tujua', 'ujuan', 'juanu', 'uanun', 'anunt', 'nuntu', 'untuk', 'ntukm', 'tukme', 'ukmen', 'kmeng', 'menge', 'engek', 'ngekp', 'gekpl', 'ekplo', 'kplor', 'plora', 'loras', 'orasi', 'rasif', 'asife', 'sifen', 'ifeno', 'fenom', 'enome', 'nomen', 'omena', 'menay', 'enaya', 'nayan', 'ayang', 'yangt', 'angte', 'ngter', 'gteru', 'terus', 'erusm', 'rusme', 'usmen', 'smeni', 'menin', 'ening', 'ningk', 'ingka', 'ngkat']
		N-Gram dengan N=7
		['penelit', 'eneliti', 'nelitia', 'elitian', 'litiani', 'itianin', 'tianini', 'ianinib', 'aninibe', 'niniber', 'inibert', 'nibertu', 'ibertuj', 'bertuju', 'ertujua', 'rtujuan', 'tujuanu', 'ujuanun', 'juanunt', 'uanuntu', 'anuntuk', 'nuntukm', 'untukme', 'ntukmen', 'tukmeng', 'ukmenge', 'kmengek', 'mengekp', 'engekpl', 'ngekplo', 'gekplor', 'ekplora', 'kploras', 'plorasi', 'lorasif', 'orasife', 'rasifen', 'asifeno', 'sifenom', 'ifenome', 'fenomen', 'enomena', 'nomenay', 'omenaya', 'menayan', 'enayang', 'nayangt', 'ayangte', 'yangter', 'angteru', 'ngterus', 'gterusm', 'terusme', 'erusmen', 'rusmeni', 'usmenin', 'smening', 'meningk', 'eningka', 'ningkat']
2.	pendidikan merupakan kuncisukses	**N-Gram with N=3**
		['pen', 'end', 'ndi', 'did', 'idi', 'dik', 'ika', 'kan', 'anm', 'nme', 'mer', 'eru', 'rup', 'upa', 'pak', 'aka', 'kan', 'ank', 'nku', 'kun', 'unc', 'nci', 'cis', 'isu', 'suk', 'uks', 'kse', 'ses']
		N-Gram dengan N=5
		['pendi', 'endid', 'ndidi', 'didik', 'idika', 'dikan', 'ikanm', 'kanme', 'anmer', 'nmeru', 'merup', 'erupa', 'rupak', 'upaka', 'pakan', 'akank', 'kanku', 'ankun', 'nkunc', 'kunci', 'uncis', 'ncisu', 'cisuk', 'isuks', 'sukse', 'ukses']
		N-Gram dengan N=7
		['pendidi', 'endidik', 'ndidika', 'didikan', 'idikanm', 'dikanme', 'ikanmer', 'kanmeru', 'anmerup', 'nmerupa', 'merupak', 'erupaka', 'rupakan', 'upakank', 'pakanku', 'akankun', 'kankunc', 'ankunci', 'nkuncis', 'kuncisu', 'uncisuk', 'ncisuks', 'cisukse', 'isukses']

F. Hash Value

At this stage, after forming the n-gram on Jaccard, the next process is looking for the hash value of each string that has been cut in each n-gram. Each string that has been cut is converted into ASCII by calculating the equation, namely:

$H(c.ck) = c1 * b (k-1) + c2 * b (k-2) + .. + ck * b (kk)$ (1)

$H(c2.ck+1)= (H(c1..ck) – c1 * b (k-1))*b + c (k+1)$ (2)

The results of rolling hash calculations with values n=3, 5 and 7 are the n-gram text results, one example of a dataset.

Text1.Source = "education is the key to success" and

Text2.Target = "learning is the foundation of success"

The following are the results of one of the calculation datasets from the presentation of rolling hash n=3:

TABLE 5
CALCULATION OF ROOLING HASH N=3 IN TEXT1 SOURCE

Text1 N=3	ASCII decimal	Calculation	Results
pen	[112,101,110]	112 * 256^(2) + 101 * 256^(1) + 110 * 256^(0)	7365998
end	[101,110,100]	101 * 256^(2) + 110 * 256^(1) + 100 * 256^(0)	6647396
ndi	[110,100,105]	110 * 256^(2) + 100 * 256^(1) + 105 * 256^(0)	7234665
did	[100,105,100]	100 * 256^(2) + 105 * 256^(1) + 100 * 256^(0)	6580580
idi	[105,100,105]	105 * 256^(2) + 100 * 256^(1) + 105 * 256^(0)	6906985
dik	[100,105,107]	100 * 256^(2) + 105 * 256^(1) + 107 * 256^(0)	6580587
ika	[105,107,97]	105 * 256^(2) + 107 * 256^(1) + 97 * 256^(0)	6908769
kan	[107,97,110]	107 * 256^(2) + 97 * 256^(1) + 110 * 256^(0)	7037294
anm	[97,110,109]	97 * 256^(2) + 110 * 256^(1) + 109 * 256^(0)	6385261

nme	[110,109,101]	110 * 256^(2) + 109 * 256^(1) + 101 * 256^(0)	7236965
mer	[109,101,114]	109 * 256^(2) + 101 * 256^(1) + 114 * 256^(0)	7169394
eru	[101,114,117]	101 * 256^(2) + 114 * 256^(1) + 117 * 256^(0)	6648437
rup	[114,117,112]	114 * 256^(2) + 117 * 256^(1) + 112 * 256^(0)	7501168
upa	[117,112,97]	117 * 256^(2) + 112 * 256^(1) + 97 * 256^(0)	7696481
pak	[112,97,107]	112 * 256^(2) + 97 * 256^(1) + 107 * 256^(0)	7364971
aka	[97,107,97]	97 * 256^(2) + 107 * 256^(1) + 97 * 256^(0)	6384481
kan	[107,97,110]	107 * 256^(2) + 97 * 256^(1) + 110 * 256^(0)	7037294
ank	[97,110,107]	97 * 256^(2) + 110 * 256^(1) + 107 * 256^(0)	6385259
nku	[110,107,117]	110 * 256^(2) + 107 * 256^(1) + 117 * 256^(0)	7236469
kun	[107,117,110]	107 * 256^(2) + 117 * 256^(1) + 110 * 256^(0)	7042414
unc	[117,110.99]	117 * 256^(2) + 110 * 256^(1) + 99 * 256^(0)	7695971
nci	[110, 99, 105]	110 * 256^(2) + 99 * 256^(1) + 105 * 256^(0)	7234409
cis	[99,105,115]	99 * 256^(2) + 105 * 256^(1) + 115 * 256^(0)	6515059
isu	[105,115,117]	105 * 256^(2) + 115 * 256^(1) + 117 * 256^(0)	6910837
suk	[115,117,107]	115 * 256^(2) + 117 * 256^(1) + 107 * 256^(0)	7566699
uks	[117,107,115]	117 * 256^(2) + 107 * 256^(1) + 115 * 256^(0)	7695219
kse	[107,115,101]	107 * 256^(2) + 115 * 256^(1) + 101 * 256^(0)	7041893
ses	[115,101,115]	115 * 256^(2) + 101 * 256^(1) + 115 * 256^(0)	7562611

G. Formation of Window and Fingerprint Values

At this stage, the formation of the window and fingerprint jaccard similarity for winnowing of the hash value obtained by grouping as many w values as possible. Following are the results of forming the window value with w = 3

d1 = "pendidikanmerupakankuncisukses"

d2 = "belajaradalahfondasikesuksesan"

N-Gram with N=3			
Windows with w=3 for d1	Windows With w=3 for d2	Fingerprints for d1	Fingerprint for d2
[0,0,0,0,0, 0.0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 3, 28, 230, 1846, 14770, 118165]	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 3, 28, 230 , 1846, 14768, 118149]	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 3, 28, 230 , 1846]	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 3, 28, 230 , 1846]
N-Gram with N= 5			
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 7, 58, 469, 3757, 30059, 240475, 1923804, 15390438, 123123510, 984988082, 7879904661]	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 7, 57, 461, 3692, 29541, 236331, 1890652, 15125222 , 121001782, 968014256, 7744114053]	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 7, 58, 469, 3757, 30059, 240475, 1923804, 15390438 , 123123510]	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 7, 57, 461, 3692, 29541, 236331, 1890652, 15125222 , 121001782]
N-Gram with N=7			
[0,0,0,0,0,0 0,0,0,0,0,1, 13,105,843, 6748.53990, 431927,345 5418, 27643349, 221146797, 1769174379 1415339503 113227 160284, 9058172 82278, 7246538 258230, 5797230 6065842, 4637784 48526741]	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 14, 117, 939, 7514, 60118, 480951, 3847609, 30780877, 246247020, 1969976165, 15759809 323 12607847458 10086277967 10 8069022373 86, 64552178989 488, 51641743191 5909]	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 13, 105, 843, 6748, 53990, 431927, 3455418, 27643349, 221146797, 176917437 9, 14153395 035, 113227160 284, 905817282 278 , 724653825 8230]	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 14, 117, 939, 7514, 60118, 480951, 3847609, 30780877, 246247020, 1969976165 , 15759809 323, 1260784745 88, 1008627796 710 , 8069022373 686]

H. TF-IDF

The dataset has gone through the data pre-processing stage, then the word weighting stage is carried out using the cosine similarity algorithm method. This stage is a stage for calculating the weight value of text similarity and the length of different n-gram values. The following are the results of weighting words based on text structure.

```
d1 = "pendidikanmerupakankuncisukses"
d2 = "belajaradalahfondasikesuksesan"
# Formation of n-grams with values N=3, 5, and 7
n_values = [3, 5, 7]
for n in n_values:
ngrams_d1 = generate_ngrams(d1, n)
ngrams_d2 = generate_ngrams(d2, n)
# Combine n-grams into sentences for vectorization
text_d1 = ' '.join(ngrams_d1)
text_d2 = ' '.join(ngrams_d2)
# Calculate and display TF-IDF in tabular form
tfidf_results = calculate_tfidf([text_d1, text_d2])
print(f"\nTF-IDF with N={n}:")
print(tfidf_results)
```

```
TF-IDF with N=3:
        ada       ahf       aja       aka       ala       ank       anm  \
0  0.000000  0.000000  0.000000  0.188898  0.000000  0.188898  0.188898
1  0.196022  0.196022  0.196022  0.000000  0.196022  0.000000  0.000000
        ara       asi       bel  ...       pen       rad       rup       san  \
0  0.000000  0.000000  0.000000  ...  0.188898  0.000000  0.188898  0.000000
1  0.196022  0.196022  0.196022  ...  0.000000  0.196022  0.000000  0.196022
        ses       sik       suk       uks       unc       upa
0  0.134402  0.000000  0.134402  0.134402  0.188898  0.188898
1  0.139471  0.196022  0.139471  0.139471  0.000000  0.000000
[2 rows x 51 columns]
TF-IDF with N=5:
adala     ahfon     ajara     akank     alahf     ankun     anmer     arada
\
0  0.00000  0.00000  0.00000  0.19995  0.00000  0.19995  0.19995  0.00000
1  0.19995  0.19995  0.19995  0.00000  0.19995  0.00000  0.00000  0.19995
asike     belaj  ...     pakan     pendi     radal     rupak     sesan  \
0  0.00000  0.00000  ...  0.19995  0.19995  0.00000  0.19995  0.00000
1  0.19995  0.19995  ...  0.00000  0.00000  0.19995  0.00000  0.19995
sikes     sukse     ukses     uncis     upaka
0  0.00000  0.142266  0.142266  0.19995  0.19995
1  0.19995  0.142266  0.142266  0.00000  0.00000
[2 rows x 50 columns]
```

```
TF-IDF with N=7:
adalahf     ahfonda     ajarada     akankun     alahfon     ankunci     anmerup  \
0  0.000000  0.000000  0.000000  0.204124  0.000000  0.204124  0.204124
1  0.204124  0.204124  0.204124  0.000000  0.204124  0.000000  0.000000
aradala     asikesu     belajar  ...     ondasik     pakanku     pendidi     radalah  \
0  0.000000  0.000000  0.000000  ...  0.000000  0.204124  0.204124  0.000000
1  0.204124  0.204124  0.204124  ...  0.204124  0.000000  0.000000  0.204124
rupakan     sikesuk     suksesa     uksesan     uncisuk     upakank
0  0.204124  0.000000  0.000000  0.000000  0.204124  0.204124
1  0.000000  0.204124  0.204124  0.204124  0.000000  0.000000
[2 rows x 48 columns]
```

The calculations obtained from two text documents (d1 and d2) depict one document and each column represents words with different n-gram lengths (N=3, N=5 and N=7). With the TF-IDF value from the calculation of one of the source text and target text datasets, it shows that the n-gram variations provide different values.

III. Results and Discussions

Based on the text dataset obtained that has gone through the preprocessing stage and the formation of n-grams on jaccard and tf-idf word weighting on cosine, the next stage will be modeling the results of the system that has been created previously. The process of implementing a comparison of the two algorithms in detecting similarities between two texts, Cosine and Jaccard coefficient results for winnowing with the Python programming language.

A. Cosine Similarity

From the two cosine and jaccard similarity algorithms obtained, the following is the algorithm model for text similarity detection based on TF-Idf representation and the n-gram hashing method on one of the datasets

d1 "education is the key to success" and

d2 "learning is the foundation of success".

The following cosine similarity results provide an illustration of the cosine value changing according to the N-gram in the text document as follows:

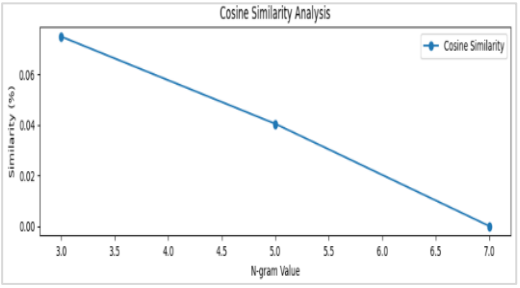

Fig. 2 Cosine similarity model

In Fig. 2. The visual cosine model is visible showing the value of each n=3,5 and 7 points on the plot marked with a circle symbol, a line plot of the N-Gram value.

B. Jaccard Similarity

The jaccard value changes according to the N-gram in the text document.

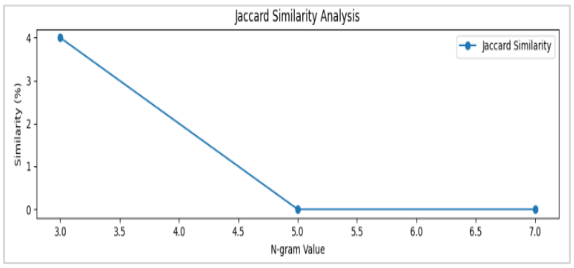

Fig. 3 Jaccard similarity model

C. Comparison of Cosine and Jaccard Similarity Results

In the results of detecting similarities between text d1 and d2, the execution time of cosine and jaccard against N-Gram from a comparison of the two algorithms shows that the time for both models changes with changes in the n-gram value.

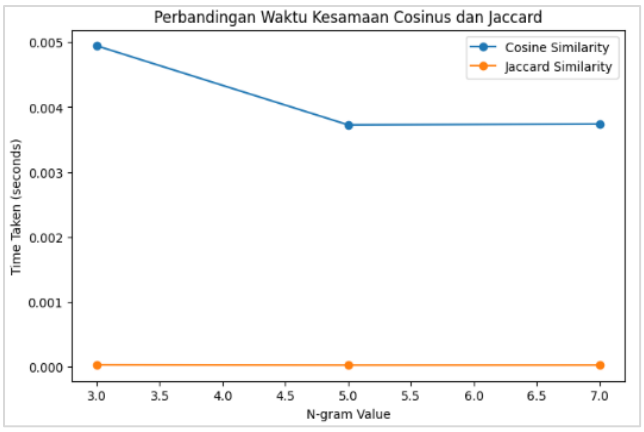

Fig. 4 Computational Time of Cosine and Jaccard Execution

With different N-Gram lengths measuring the time of each model, listed with time_taken_cosine and time_taken_jaccard, with visual plot results you can compare speed and efficiency between models for each N-Gram value in the dataset

(d1 "education is the key to success" and
d2 "learning is the foundation of success").

TABLE 6
EXAMPLE OF DATASET SOURCE DATA AND TARGET DATA

N-grams	Cosine Similarity	Cosine Similarity Time	Jaccard Similarity	Jaccard Similarity Time
3	0.074%	0.004944	4.0%	0.000029
5	0.040%	0.003726	0.0%	0.000025
7	0.0%	0.003740	0.0%	0.000026

In the process of implementing the text-based plagiarism detection system in this research between texts

Fig. 5 Test results display

In Fig. 5. shows the interface of the implementation of the similarity detection algorithm between texts, to compare the texts to be compared, you can input the source in the first textarea column, then input the text to be compared in the second textarea, the n-gram values, prime numbers and window can be adjusted to your needs. To check the similarity between texts, the user clicks the submit detection button on the system. The following is a test of similarity detection results on sample dataset data as follows:

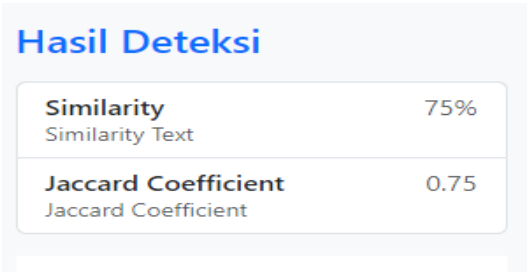

Fig. 6 Similarity Detection Test Results in the System

D. Analysis of Cosine and Jaccard Algorithm System Testing Results

The system used in testing this data was built using the Python programming language and its supporting libraries. The first test was carried out with a functionality test and compared with commonly used flows. The test data uses sentence data as in Table IV.3 Stop Word Spaces Between Characters, totaling 20 datasets. The following are the results of testing scenarios for different N-Gram variations, calculating the text similarity cosine similarity and jaccard similarity, presented in table IV.10 as follows:

TABLE 7
TEST RESULTS FOR N-GRAM COSINE AND JACCARD SIMILARITY VARIATIONS

No	Mark N-grams	Cosine Similarity	Jaccard Similarity	Amount N-Grams	Accuracy
1	3	0.45%	36.95%	37	41.57 %
	5	0.37%	32.6%	34	36.95 %
	7	0.32%	28.26%	30	32.6%
2	3	0.07%	4.0%	4	7.84%
	5	0.04%	0.0%	2	4.0%
	7	0.0%	0.0%	0	0.0%
3	3	0.53%	44.64%	27	50.94 %
	5	0.44%	37.5%	25	44.64 %
	7	0.37%	30.35%	21	37.5%
4	3	0.68%	46.15%	24	64.86 %
	5	0.48%	35.89%	18	46.15 %
	7	0.37%	35.89%	18	46.15 %
5	3	0.62%	44.44%	22	52.38 %
	5	0.45%	35.55%	20	44.44 %
	7	0.36%	26.66%	16	35.55 %
6	3	0.13%	10.0%	10	14.49 %
	5	0.09%	7.24%	7	10.0%
	7	0.07%	4.47%	5	7.24%
7	3	0.09%	4.91%	5	8.06%
	5	0.05%	1.69%	3	4.91%
	7	0.01%	0.0%	1	1.69%
8	3	0.14%	7.35%	8	12.12 %
	5	0.07%	4.41%	5	7.35%
	7	0.45%	1.51%	3	4.41%
9	3	0.27%	14.0%	15	25.0%
	5	0.14%	7.81%	9	14.0%
	7	0.07%	3.17%	5	7.81%
10	3	0.31%	29.7%	16	33.3%
	5	0.30%	26.6%	14	29.78 %
	7	0.27%	23.25%	12	25.6%
11	3	0.31%	20.0%	13	30.23 %
	5	0.20%	11.1%	9	20.0%
	7	0.11	6.97%	5	11.1%
12	3	0.06%	3.38%	4	6.55%
	5	0.03%	0.0%	2	3.38%
	7	0.0%	0.0%	0	0.0%
13	3	0.47%	42.85%	22	44.0%
	5	0.43	40.42%	21	42.85 %
	7	0.40%	37.7%	19	40.42 %
14	3	0.25%	20.68%	15	27.7%
	5	0.19%	17.54	12	20.68 %
	7	0.17%	14.54%	10	17.54 %
15	3	0.24%	14.81	13	24.52 %
	5	0.14%	7.40%	8	14.81 %
	7	0.07%	0.0%	4	7.40%
16	3	0.41%	34.69%	19	40.42 %
	5	0.35%	31.91%	17	34.69 %
	7	0.32%	28.8%	15	31.91 %
17	3	0.35%	36.73%	20	41.6%
	5	0.31%	32.65%	18	36.73 %
	7	0.31%	29.16%	16	32.65 %
18	3	0.06%	1.56%	4	6.25%
	5	0.01%	0.0%	1	1.56%
	7	0.0%	0.0%	0	0.0%
19	3	0.23%	16.98%	11	20.75 %
	5	0.17%	13.72%	9	16.98 %
	7	0.13	10.20	7	13.72 %
20	3	0.99%	100%	33	100%
	5	0.99%	100%	31	100%
	7	0.99%	100%	29	100%

The following is a visualization plot image of N-gram values depicted in graphical form.

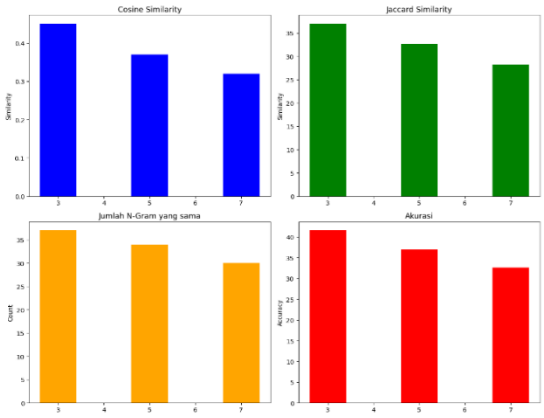

Fig. 7 Graphs of N-Gram Plot Visualization Results

In table. 3. Calculation results of similarity detection between texts using the Cosine Similarity and Jaccard Similarity algorithms, displaying the results of the percentage of text similarity from different N-Gram values. The results of both algorithms provide quite good results in detecting similarities between texts.

Based on the average accuracy results for the entire dataset, cosine similarity is better at detecting similarities between texts. Cosine and Jaccard algorithm analysis was carried out by looking at the results of checking the similarity of the data produced with various n-gram values. The accuracy value of cosine similarity tends to be higher than Jaccard similarity for most datasets in the form of continuous values between 0% to 100% (except for the 7th dataset) however, stability does not reach 100% for each dataset. Meanwhile, Jaccard similarity calculates similarity based on the resulting set of unique elements (n-grams), the accuracy value even reaches 0%, which shows that there are significant differences in similarity detection results, less stability with greater variations between datasets.

E. Confusion Matrix

Accuracy calculations are carried out using the similarity results obtained in Table IV.10 System Test Results from research and similarity results obtained by Plagiarism Checker-X software. In the process of calculating accuracy, the researcher used one of the results of the highest level of accuracy from the n-gram variation in the dataset table IV.10, measuring the light similarity level below 20% while measuring heavy similarity from 25-100%, following the results of the test data for both systems.

TABLE 8
SYSTEM COMPARISON RESULTS WITH PLAGIARISM CHECKER X TOOLS

No	Researcher System			Plagiarism Checker		
	Accuracy	Light	Heavy	Accuracy	Light	Heavy
1	41.57%		✓	11%	✓	
2	7.84%	✓		0.0%	✓	
3	50.94%		✓	0.0%	✓	
4	64.86%		✓	0.0%	✓	
5	52.38%		✓	0.0%	✓	
6	14.49%	✓		0.0%	✓	
7	8.06%	✓		0.0%	✓	
8	12.12%	✓		0.0%	✓	
9	25.0%		✓	0.0%	✓	
10	33.3%		✓	0.0%	✓	
11	30.23%		✓	0.0%	✓	
12	6.55%	✓		0.0%	✓	
13	44.0%		✓	0.0%	✓	
14	27.7%		✓	0.0%	✓	
15	24.52%		✓	0.0%	✓	
16	40.42%		✓	0.0%	✓	
17	41.6%		✓	0.0%	✓	
18	6.25%	✓		0.0%	✓	
19	20.75%	✓		0.0%	✓	
20	100%		✓	100%		✓

From the results of the table obtained, it is known that the classification table with confusion matrix is as follows:

		Plagiarism Checker	
		Light	Heavy
System Researcher	Light	7	13
	Heavy	19	1

In the confusion matrix table to calculate the classification accuracy test of the researcher's system as (actual) with the plagiarism checker tool x as (predicted) consists of four values, namely True Positive (TP) The number of similarity detections that are actually light and predicted to be light (7). False Positive (FP) The number of similarity detections that are actually heavy, but predicted to be light (1). True Negative (TN) The number of similarity detections that are actually heavy and predicted heavy (13). False Negative (FN) The number of similarity detections that are actually light, but predicted to be heavy (19). From the confusion matrix we can calculate accuracy. The following is the accuracy calculation equation as follows:

Accuracy = (TP + TN) / (TP + TN + FP + FN)
= (7 + 1 3) / (7 + 13 + 1 + 19)
= (20) / (40)
= 0.5 * 100%
= 50%

F. Results Visualization

From the results of the implementation of the comparison of the cosine and Jaccard similarity algorithms in detecting similarities between texts based on variations in N-gram values 3, 5 and 7 from a total of 20 (twenty) total datasets in this study, the following visualization of the results of the similarity level comparison is presented in the following image:

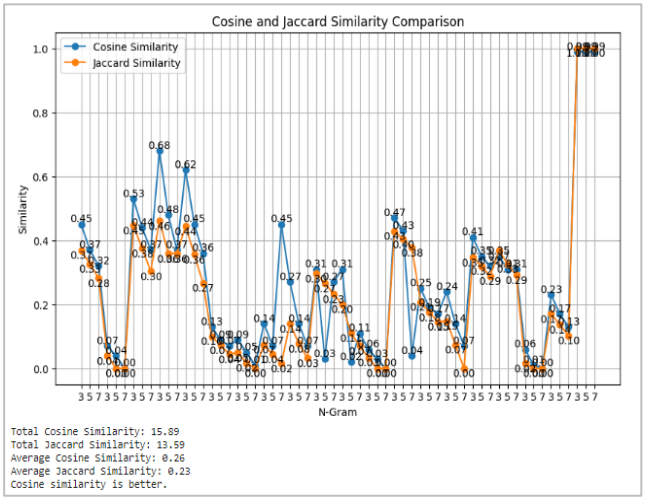

Fig. 8 Visualization of Cosine and Jaccard Similarity Comparison Results

Based on the calculation results of the similarity level comparison between the cosine similarity and jaccard similarity algorithms for detecting text similarity related to n-gram

variation values, it shows that the cosine similarity algorithm is better at detecting similarities between texts than jaccard similairty. The total performance cosine similarity is 15.89% and the average is 0.26%. Meanwhile, the total performance of Jaccard similarity was 13.59% and the average was 0.23%.

IV. Conclusions

Based on the results of tests carried out on the implementation of the system, it can be seen that detecting plagiarism by comparing the level of similarity between the cosine similarity and jaccard similarity algorithms for detecting text similarity related to n-gram variation values shows that the cosine similarity algorithm is better at detecting similarities between texts than jaccard similairty. The total performance cosine similarity is 15.89% and the average is 0.26%. Meanwhile, the total performance of Jaccard similarity was 13.59% and the average was 0.23%.

References

[1] MA Shadiqi, "Understanding and Preventing Plagiarism in Writing Scientific Papers," *Bul. Psychol.* , vol. 27, no. 1, p. 30, 2019, doi: 10.22146/buletinpsychologi.43058.

[2] F. Marsha, "Literature Review: Plagiarism Detection System in Documents Using the Rabin-Karp Algorithm Lecturer: Viktor Handrianus Pranatawijaya, ST., MT. Group Members: Kiplianor Farhan Marsha Rasyid Noor Imamsyah Jakkirahman," no. May, 2021.

[3] ZM Pahlevi R, "Designing Similarity Detection in Abstracts of Informatics Scientific Articles.pdf," vol. 8, No. 2, 2022.

[4] DD Dwi Yulianto, L., Gata, W., Frieyadie, & Saputra, "Validation of National University Research and Community Service Submission Documents Using the Finite State Automata Method," pp. 497–504, 2022, doi: https://doi.org/10.35870/jtik.v6i4.520.

[5] A. Mubarak, "Implementation of the Rabin-Karp Algorithm for Detecting Plagiarism in Web-Based Text Document Files," *J. Inf. Syst. Res.* , vol. 3, no. 3, pp. 150–154, 2022, doi: 10.47065/josh.v3i3.1404.

[6] Yuda, "Designing a Translation Application for the Holy Qur'an Using the Winnowing.pdf Algorithm," *Ilm. Inform. and Comput.* , vol. 2, No. 5, 2022, doi: https://doi.org/10.30865/klik.v2i5.361.

[7] D. Dhamayanti and LP Sari, "Plagiarism Detection Application at Indo Global Mandiri University Based on Web," *J. Ilm. Inform. Globe.* , vol. 10, no. 2, 2019, doi: 10.36982/jiig.v10i2.864.

[8] AE Budiman and A. Widjaja, "Analysis of the Effect of Text Preprocessing on Plagiarism Detection in Final Project Documents," *J. Tek. Inform. and Sis. Inf.* , vol. 6, no. 3, pp. 475–488, 2020, doi: 10.28932/jutisi.v6i3.2892.

[9] RT Wahyuni, D. Prastiyanto, and E. Supraptono, "Application of the Cosine Similarity Algorithm and TF-IDF Weighting in the Thesis Document Classification System," *J. Tek. Electrical Univ. Semarang State* , vol. 9, no. 1, pp. 18–23, 2017, [Online]. Available: https://journal.unnes.ac.id/nju/index.php/jte/article/download/10955/6659

[10] S. Sunardi, A. Yudhana, and IA Mukaromah, "Implementation of Plagiarism Detection Using the N-Gram and Jaccard Similarity Methods for the Winnowing Algorithm," *Transmission* , vol. 20, no. 3, p. 105, 2018, doi: 10.14710/transmisi.20.3.105-110.

[11] W. Desena and A. Solichin, "Searching for Student Final Project Abstracts Based on the Level of Similarity Using the Winnowing and Jaccard Similarity Algorithms at Budi Luhur University," *Inform. J. Computer Science.* , vol. 17, no. 2, p. 112, 2021, doi: 10.52958/iftk.v17i2.3628.

[12] PYE Nasien D, "Implementation of the Winnowing Algorithm in Detecting Plagiarism in Title and Abstract of Student's Final Project." pp. 2528–4061, 2022. doi: 10.25199/itjrd.2022.7879.

[13] E. da Costa and VS Mali, "Tetun Language Plagiarism Detection With Text Mining Approach Using N-gram and Jaccard Similarity Coefficient," *Timor-Leste J. Eng. Sci.* , no. January, 2021, [Online]. Available: https://www.tljes.org/index.php/tljes/article/view/9%0A https://www.tljes.org/index.php/tljes/article/download/9/7