IJAIT (International Journal of Applied Information Technology)
http://journals.telkomuniversity.ac.id/ijait/

Design and Implementation of a Final Project Plagiarism Detection System Using Cosine Similarity Method

Rival Fauzi [a, *], Muhammad Iqbal [a], Tita Haryanti [a]

[a] *Diploma of Telecommunications Technology, Telkom University, Indonesia*
rivalfauzi@student.telkomuniversity.ac.id, miqbal@telkomuniversity.ac.id, tharyanti@telkomuniversity.ac.id,

ARTICLE INFO

Received September 6th, 2021
Revised March 13th, 2022
Accepted March 15th, 2022
Available online June 24th, 2022

Keywords
plagiarism, cosine similarity,
external plagiarism detection

ABSTRACT

Plagiarism is an act of taking ideas, taking research results, acquiring research results, and summarizing writing without mentioning the source, either intentionally or unintentionally. The cosine similarity method can be used to calculate the score of similarities between documents by comparing existing documents on the database with uploaded ones to gain a similarity percentage. External plagiarism detection (EPD) is used to compare the contents of articles. The design and implementation of the system will be carried out in several stages in the hope that the system can work optimally and detect text similarities accurately. This research objective is to check the plagiarism rate of the document using the cosine similarity method as a method of calculating word equations called Kipcheck. The purpose of checking the level of plagiarism is to ensure that the documents created have a minimum level of fraud to avoid academic sanctions. Kipcheck uses three application system tests; maximum word calculation that can be processed; comparing application and manual calculation; and testing the consistency of application calculation results based on two different schemes.

Acknowledgment

Thanks to the Diploma of Telecommunication Technology, Telkom University, for providing the opportunity to implement the results of this research.

* Corresponding author at:
School of Applied Science, Telkom University,
Jl. Telekomunikasi No. 1, Terusan Buah Batu, Bandung, 40257
Indonesia.
E-mail address: rivalfauzi@student.telkomuniversity.ac.id

ORCID ID:
- First Author: 0000-0002-5619-1817
- Second Author: 0000-0001-6508-8283

https://doi.org/10.25124/ijait.v5i02.4146
Paper_reg_number IJAIT000050201 2022

1. Introduction

Plagiarism is not allowed in any academic activity. In addition, plagiarism is also contrary to the honesty needed in the scientific and academic world. Without honesty, science would not have evolved as it is today. However, there are still many people engaged in science, both as academics and researchers, committing acts of plagiarism. One of the causes of this is the lack of ability to write scientific articles.

Determining plagiarism in writing is not easy because comparison documents are very much, especially in today's digital era. Currently, many articles are written in cyberspace. Therefore, an application is needed to help detect the potential plagiarism of writing.

Irfan Pahlevi et al. [1] once developed a web application that can calculate the similarity level of an abstract of a student's final task with another final task abstract using the cosine similarity method, proving that the cosine similarity method can be used to detect plagiarism. Cosine similarity has also been used by Gunawan et al. [2] to find relationships between the text of the two documents. In addition, research conducted by Tomáš Foltýnek et al. [3] has spelled out several points of hope for developing plagiarism detection applications, one of which is the detection of using words that have the same meaning or synonyms of words. [4] [5]

Several online sites have developed their respective detection systems, from open-source to paid ones. Sites smallseotools.com [6] have tools used to detect plagiarism, with different types of input documents that can be used for free. However, detection is limited to 1000 words without login and 5000 words by the login. If it exceeds the word limit, there will be an error. The unicheck.com [7] site also has a plagiarism detection system with output in percentages and hyperlinks to sources that are considered genuine, but money is required for subscription fees. Services on duplichecker.com [8] can detect other online site inputs by placing hyperlinks to the site to be checked, but full use of the service is required to pay a subscription fee. Based on some online site references, it can be concluded that the sameness of various plagiarism detection online sites is only to reach documents uploaded on the internet and not encrypted.

This research designed a website-based plagiarism detection system called Kipcheck. The system will then be applied to the final project Diploma of Telecommunication Technology database to restrict it to local databases. The comparison document that will be used as a reference is a document in the application database and not a document on the internet.

Systematic writing in this research is as follows: section (1) describes the background of making the Kipcheck; part (2) describes the system model, user interface design, and calculation analysis of Kipcheck; section (3) describes the results of maximum word testing, numeracy accuracy testing, and Kipcheck consistency testing; and section (4) describes the conclusions of the Kipcheck analysis.

2. Literature

2.1. Plagiarism

According to the KBBI, plagiarism is taking other people's writings (opinions, etc.) and making them look like their own compositions (opinions, etc.), for example, publishing other people's writings on their behalf [9]. In contrast, plagiarism itself is an act of plagiarism that violates copyright [10]. The explanation

of plagiarism in Indonesian law is contained in the regulation of the Minister of National Education number 17 of 2010 chapter I article one [11]. In addition, regulations regarding sanctions for acts of plagiarism are written in law number 20 of 2003, article 25 paragraph 2 [12]. There are several categories of plagiarism, namely word by word plagiarism, Word switch plagiarism, Metaphor plagiarism, Idea plagiarism, and self-plagiarism [13][14].

2.2. Cosine Similarity

Cosine similarity is a method to calculate the similarity between two objects expressed in two vectors using the keyword of a document as a measure [15]. Cosine Similarity will do the math compare (similarity) to the comparison between two or more objects. Results from the calculation in the form of a cosine angle x with 0 (zero) as the smallest value, which means it has no value and 1 (one) as the largest value contained Mark [16].

3. System Model

3.1. Kipcheck System Model

This research designed and implemented a website-based document plagiarism rate calculation application. This app is called Kipcheck. The Kipcheck system model is shown in Figure 1.

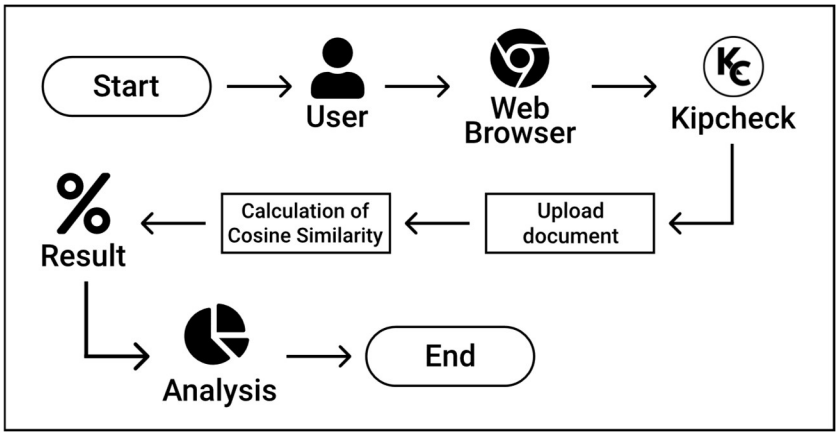

Figure 1 Kipcheck System Model

A user is required to use one of the web browsers as a tool to access Kipcheck. Then the user is asked to enter data in the form of documents that will be checked for plagiarism. Furthermore, the calculation of word similarities using the cosine similarity method will be done automatically by Kipcheck. Once the calculation is complete, Kipcheck will display results that the user can analyze. Results displayed by Kipcheck in the form of (i) Blocking text and (ii) plagiarism percentage.

3.2. The Overall Process of Work

This research designed and supplemented the application of plagiarism level calculation in the final project document to make it easier for students and lecturers to detect and monitor the final project document related to the level of plagiarism. The design stages of this research are represented in diagrams in Figure 2.

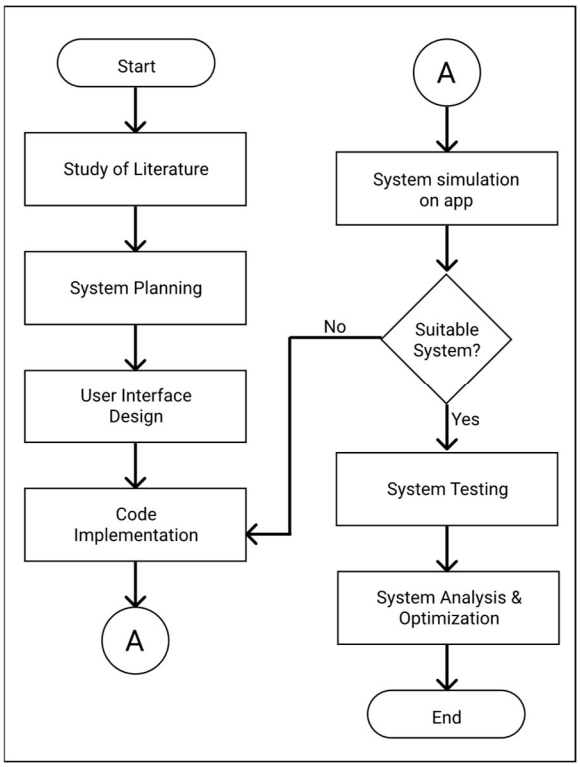

Figure 2 Design Stage

3.3. User Interface Design

The user interface design is a display implemented in Kipcheck. User interface design is created with the aim that users can interact with the built application. Here is the interface found in Kipcheck.

Figure 3 Design of The Main Page

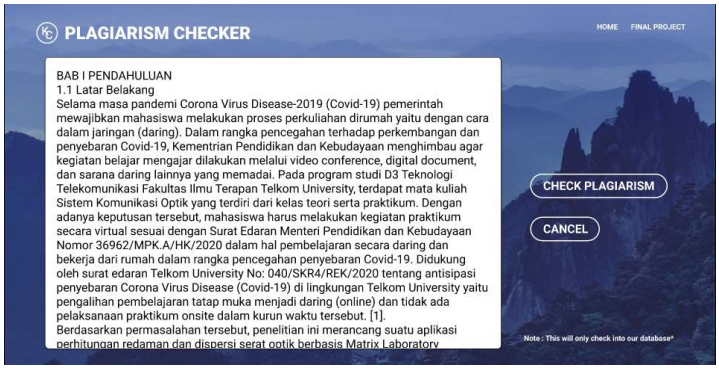

Figure 4 Design of The Index Page

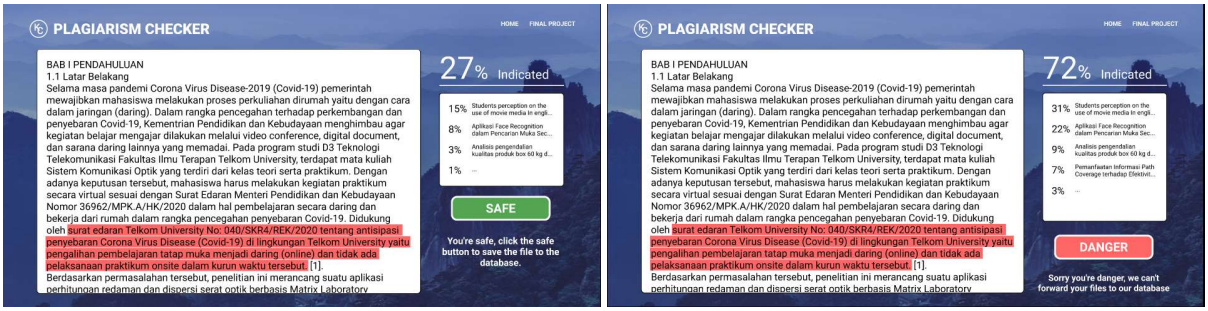

Figure 5 Design of The Results Page

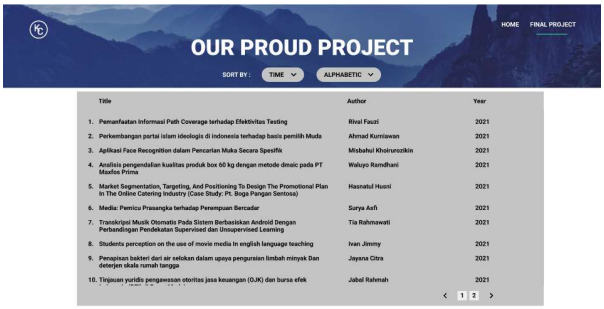

Figure 6 Design of The Final Project Page

Figure 3 shows the design of the main page that the user uses to enter the document. Figure 4 shows the design of an index page that is useful for correcting whether the entered document is correct. Figure 5 shows the design of the results page divided into two conditions, the first condition for documents that have below-grade results and the second condition for documents that have results above-grade. Figure 6 shows the final project page design to display the entire final project document in the database.

3.4. Manual Calculation of Cosine Similarity

Manual calculations are done with the help of Microsoft Excel as a tool for calculating and collecting words. The formula for calculating cosine similarity is as shown in Equation 1.

$$\frac{\sum_{n=1}^{j}(nA\ x\ nB)}{\sqrt{\sum_{n=1}^{j}(nA)^2}\ x\ \sqrt{\sum_{n=1}^{j}(nB)^2}} \qquad \text{Equation 1}$$

The value of j is the absolute value of A ∩ B. nA is the number of occurrences of the index word (n) th of the word list in sentence A. Then, the value of nB is the number of occurrences of the index word (n)th from the list of words in sentence B.

The steps for calculating cosine similarity are shown in Figure 7.

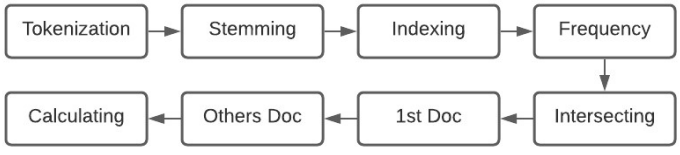

Figure 7 Steps for Calculating Cosine Similarity

1. Tokenization

   Tokenization is the process of taking all the words in a document. The output result of the tokenization stage is the same data like the contents of

the document. Contributed to drafting the article or reviewing and revising it for intellectual content

2. Stemming
   Stemming is the process of adjusting abbreviations, converting all letters of a word into lowercase, and eliminating symbols. Input data from the steaming stage is data from tokenization, which is then processed and produces sentences that use the basic word of each word in the document.
3. Word Indexing
   Word indexing is breaking sentences into words, and each word is gathered in a set of words. When there are two or more of the same words in the entire document, the word is saved only once. Input data at this stage of word indexing uses output data from the steaming stage, so the content of the word set is the basic word of each word.
4. Determining Term Frequency
   The term frequency (TF), which will next be called F is the number of occurrences of words in a document. The list of words used to define F is a collection of words that have been collected at the indexing stage of the word. The result of determining F is a table that lists the words F in the first document and F in the second document.
5. Determine the number of intersecting word values
   This stage is the process of determining the numerator in the formula shown in Equation 1. Determining the number of word values that contain this by summing the entire multiplication between F in the first document and F on the second document, each word is the same.
6. Determine the overall value of the word in the first document
   This stage is the process of determining one of the denominator components in the formula shown in Equation 1. The determination is made by entrenching the total number of F in the first document of two.
7. Determine the overall value of the word in the second document
   Almost the same as stage 6, this stage is the process of determining one of the denominator components in the formula indicated by Equation 1. The difference lies in the calculated document, at stage 6 using the first document, whereas it uses the second document.
8. Calculating Cosine Similarity
   The calculation of cosine similarity uses the formula in Equation 1, with the numerator predetermined at stage 5 as well as the denominator at stages 6 and 7.

4. Results of Discussion

4.1. Kipcheck Design Results

The final result of designing this application is an application that is used to calculate the plagiarism rate of the final project using the cosine similarity method.

Figure 8 Implementation of the main page.

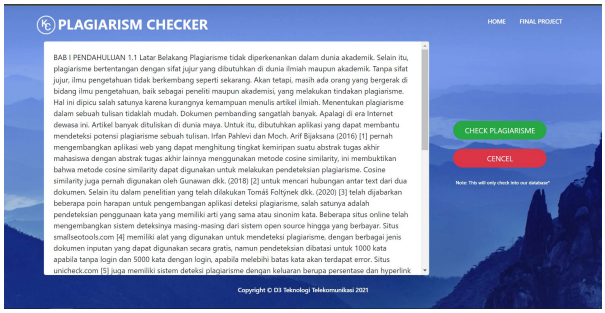

Figure 9 Implementation of the index page.

Figure 10 Implementation of the result page.

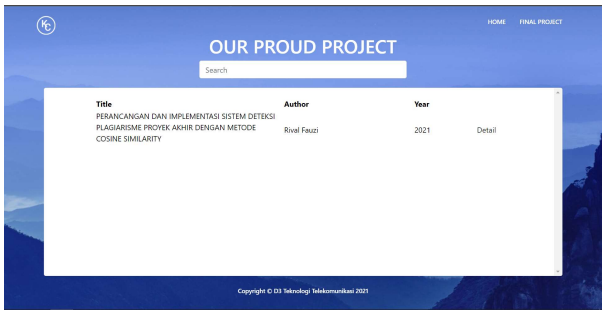

Figure 11 Implementation of the final project page.

Figure 8 shows the implementation of the main page, and the display will ask the user to upload documents. Figure 9 shows the implementation of the index page, displaying the contents of the document previously entered by the user on the main page. Figure 10 shows the implementation of the results page with two different page implementations. The percentage of plagiarism results is above grade, and on the right, the percentage of plagiarism results is below grade. Figure 11 shows the implementation of the final project page, and the system will display all documents in the database.

4.2. Deployment System

The deployment system is intended so that the system of the Kipcheck can be accessed anywhere and anytime. This Final Project deploys on Google compute engines with virtual machine specifications, as shown in Table 1.

Table 1 Virtual Machine Specifications

Hosting Specifications	Information
Cloud Platform	Google Cloud Platform
IP	http://34.101.136.165/
Machine Type	e2-medium
Disk Storage	4 GB memory
CPU	2 vCPUs
OS	Ubuntu 20.04 LTS

This research uses several features provided by Google, Google Drive API, and Google Spreadsheet API as the database. Google Drive API is useful for saving source documents in pdf file form. At the same time, Google Spreadsheet API is used to connect Kipcheck with google spreadsheet, which is useful for storing data in text form. The table's contents in the spreadsheet can be seen in Table 2.

Table 2 Contents of the Spreadsheet Table

No	Table Name	Column Content
1	Id	Contains identification codes from data and documents that have been stored in the database
2	Title	Contains the title of the input document stored in the database
3	author	Contains the author's name of the input document stored in the database
4	Nim	Contains the student ID number from the author of the input document
5	lecture1	Contains the name of the lecturer 1
6	lecture2	Contains the name of the lecturer 2
7	Year	Contains the year of manufacture of the input document
8	File	Contains a google drive link for the input document
9	created_at	Contains data creation time data

Documents and data in the database are processed manually, and admins need to make changes in google drive and google spreadsheets. If admins want to enter the source document into the database, an admin must upload the pdf document to google drive and enter the text data into the google spreadsheet. In this way, the source documents for this research are compiled. To change the data, the admin only needs to change the contents of the columns in the google spreadsheet. The source document in the database will continue to be used as the source document. To delete it, the admin must delete the document in google drive and the data in the google spreadsheet.

4.3. Maximum Word Testing

Maximum word testing is done to determine the number of words that the library can process. In this maximum word testing, it is known that the library can process words up to 14938 words contained in the input document. The number of

words is not affected by the number of words in the document in the database. If it exceeds the number of words limit, then the library cosine similarity will error. This test is done by adding words to the input document.

4.4. Comparative Analysis of Cosine Similarity Calculations

Analysis of cosine similarity calculations is done by comparing manual calculation simulations and calculation simulations in the Kipcheck. Manual calculations are used as a comparison because manual calculations are calculations that yield true values. This analysis was conducted to determine the level of accuracy of the automatic cosine similarity calculation on Kipcheck. The analysis was conducted using four different sentences that represent the document, with one sentence as a comparison sentence for another sentence, the sentence of which is:

Q : "D3 Teknologi Telekomunikasi Universitas Telkom terakreditasi A",
D1: "Universitas Telkom adalah universitas swasta terbaik di Indonesia",
D2: "Mahasiswa D3 Teknologi Telekomunikasi tampan-tampan" dan,
D3: "Universitas Telkom terakreditasi BANPT"

The manual calculation stage of cosine similarity is carried out by point 2.4. Manual calculations are done with Microsoft Excel to help calculate and collect words in table form.

Analysis of the results of cosine similarity calculations in the Kipcheck only uses a back-end system because of the simplicity and ease of testing. In addition, the back-end system has represented the calculation of cosine similarity in the Kipcheck. Input variables in the form of strings and arrays, shown in the source code.

```
const q = 'D3 Teknologi Telekomunikasi Universitas Telkom terakreditasi A';
const d = [
        'Universitas Telkom adalah universitas swasta terbaik di Indonesia',
        'Mahasiswa D3 Teknologi Telekomunikasi tampan-tampan',
        'Universitas Telkom terakreditasi BANPT'
];
```

For example, the calculation of cosine similarity between document q and the first sentence in document d is carried out based on the steps in section **Error! Reference source not found.** as follows.

1. Tokenization
   Tokenization result as follows:
   q : D3 Teknologi Telekomunikasi Universitas Telkom terakreditasi A
   d : Universitas Telkom adalah universitas swasta terbaik di Indonesia
2. Stemming
   Stemming result as follows:
   q : d3 teknologi telekomunikasi universitas telkom akreditasi a
   d : universitas telkom adalah swasta baik di Indonesia
3. Word Indexing, the stored word sets are shown in Table 3.

Table 3 Word Indexing Result

Words
d3
teknologi
telekomunikasi
universitas

Words
Telkom
akreditasi
A
adalah
swasta
baik
di
indonesia

4. Determining Term Frequency, the results of the determination F are shown in Table 4.

Table 4 Result of Determining Term Frequency

Word	Fq	Fd
d3	1	0
teknologi	1	0
telekomunikasi	1	0
universitas	1	2
Telkom	1	1
akreditasi	1	0
A	1	0
adalah	0	1
swasta	0	1
baik	0	1
di	0	1
indonesia	0	1

5. Determine the number of intersecting word values, the results at this stage are shown in Table 5.

Table 5 Result of Determining the Number of Intersecting Word Values

Word	Fq	Fd	Fq × Fd
d3	1	0	0
teknologi	1	0	0
telekomunikasi	1	0	0
universitas	1	2	2
telkom	1	1	1
akreditasi	1	0	0
A	1	0	0
adalah	0	1	0
swasta	0	1	0
baik	0	1	0
di	0	1	0
indonesia	0	1	0
	$\sum_{n=1}^{j}(nA\ x\ nB)$		3

The formula in Table 5 is obtained from equation 1 in the numerator.

6. Determine the overall value of the word in the first document, the results at this stage are shown in Table 6.

Table 6 Result of Determining the Overall Value of the Word in the First Document

Word	Fq	Fd	$(Fq)^2$
d3	1	0	1
Teknologi	1	0	1
Telekomunikasi	1	0	1
Universitas	1	2	1
Telkom	1	1	1
Akreditasi	1	0	1
A	1	0	1
Adalah	0	1	0
Swasta	0	1	0
Baik	0	1	0
Di	0	1	0
Indonesia	0	1	0
	$\sqrt{\sum_{n=1}^{j}(nA)}$		2.645751311

The formula in Table 6 is obtained from equation 1 in the first denominator.

7. Determine the overall value of the word in the second document. The results at this stage are shown in Table 7.

Table 7 Result of Determining the Overall Value of the Word in the Second Document

Word	Fq	Fd	$(Fd)^2$
d3	1	0	0
teknologi	1	0	0
telekomunikasi	1	0	0
universitas	1	2	4
telkom	1	1	1
akreditasi	1	0	0
A	1	0	0
adalah	0	1	1
swasta	0	1	1
baik	0	1	1
di	0	1	1
indonesia	0	1	1
	$\sqrt{\sum_{n=1}^{j}(nB)}$		3.16227766

The formula in Table 7 is obtained from equation 1 in the second denominator.

8. Calculating Cosine Similarity. Using equation 1, the cosine similarity calculation results from 0.3585686, the cosine angle between 0 (zero) and 1 (one). To convert the percentage result, it must be multiplied by 100% so that the results of the similarity are 35.8585686%. The same steps are carried out on document q with other sentences in document d. Comparing cosine similarity calculations between manual and automatic calculations is shown in Table 8.

Table 8 Comparison of Cosine Similarity Calculations

Information	Manual results	Application results
Q dan D1	35,8585686%	35,86%
Q dan D2	40,0891863%	40,09%
Q dan D3	56,694671%	56,69%

Calculation of total accuracy value (A_t) aims to ensure manual calculations and application calculations produce accurate values. The calculation of the value of A_t Obtained by calculating each parameter i Equation 2.

$$A_t = \sum_{i=1}^{n} \frac{1}{n} A_i \times 100\% \qquad \text{Equation 2}$$

The calculation of the total accuracy value (A_t) aims to ensure that manual calculations and application calculations produce accurate values. With A_t is the total accuracy value in the form of percentages, A_i is the i-point accuracy value, and n represents the number of A_i. The stages of calculating the percentage value of the (i)th cosine similarity accuracy (A_i) are carried out by calculating the data (D), the average data (R), the difference in the average data value (δ), the average error of each measurement from the actual value (Y), and the error percentage (E), as follows.

1. Determine the value of D;
2. Determine the value of R, which is the average value of the data obtained from D;
3. Determine the value of δ, that is, by subtracting the value of D by R;
4. Determine the value of Y, that is, by dividing the sum of the absolute values of δ;
5. Determine the value of E, by multiplying the value of Y by 100% to get the percentage of the error value;
6. Then after getting the value of E, the value of A_i It can be obtained by subtracting the value of 100% from the value of E that has been obtained.

The following is an example of calculating accuracy with a D value obtained from Table 8.

$$D = 35.8585686; 35.86$$
$$R = 35.8592843$$
$$\delta = D - R$$
$$= -0.0007157; 0.0007157$$
$$Y = \frac{|0.0007157 + 0.0007157|}{2} = 0.0007157$$
$$E = Y \times 100\%$$
$$= 0.0007157 \times 100\%$$
$$= 0.07157\%$$
$$A_1 = 100\% - E$$

$$= 100\% - 0.07157\%$$
$$= 99.9284\%$$

The same way is done repeatedly as much as the data contained in the table produces $A_1 = 99.9284\%$, $A_2 = 99.959315\%$, and $A_3 = 99.76645\%$.

Based on Equation 2 with the parameter A_i that has been obtained, that is

$$A_i = 99.9284\% + 99.959315\% + 99.76645\%$$
$$= 2.99654165$$

$$A_t = \sum_{i=1}^{3} \frac{1}{3} 2.99654165 \times 100\% = 99.884721667\%$$

Results from the total accuracy value (A_t) is 99.884721667%, stating that the Kipcheck has a near-perfect accuracy value.

4.5. Testing of Cosine Similarity Consistency Results

Testing is done with two schemes, and schema-1 calculates cosine similarity between the input document and the document in the database after all the contents of all documents are put together in one document. While schema-2, the cosine similarity calculation is done between the input documents with each document in the database one by one, then the highest similarity value is selected due to the calculation. The test was conducted with three input documents, namely the first, second, and third documents that had a word count of 500 words, 375 words, and 625 words, in a row, as shown in Figures 12.

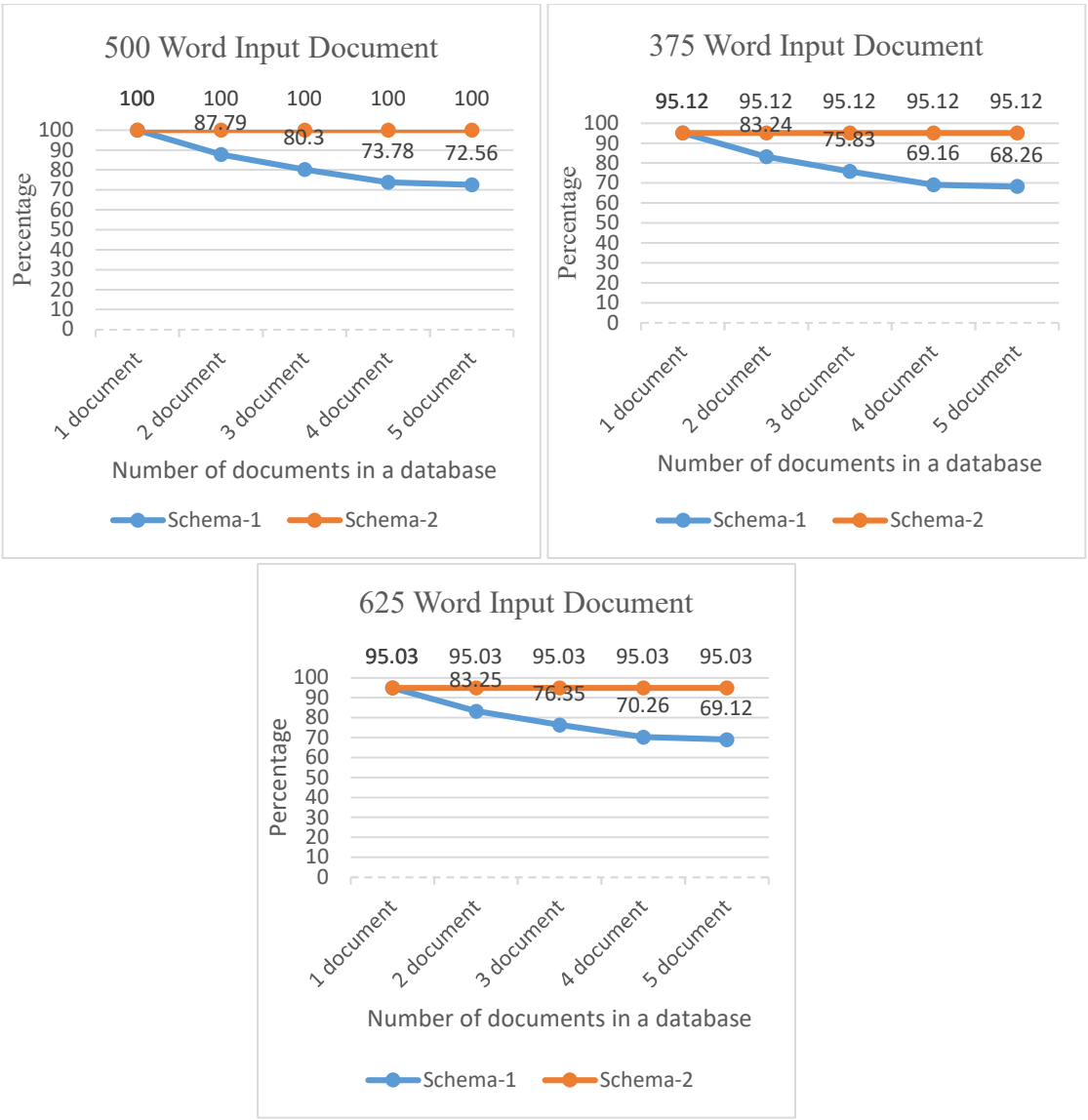

Figure 12 Result of CONSISTENCY TESTING.

The x-axis represents the number of source documents in the database. At the same time, the y-axis is the result of calculating the cosine similarity between the input document and the document in the database. Each time a calculation experiment is performed on a schema with the same input document, a new source document is added to the database.

Based on the graph results in Figure 12, it can be concluded that the results of similarity in schema-1 are inversely proportional to the number of documents in the database. The more number of documents in the database, the similarity results of schema-1 will decrease. At the same time, schema-2 has consistent similarity results and does not depend on the number of documents in the database. Therefore, schema-2 is the best scheme in this Final Project to be applied to the Kipcheck because of the consistent calculation results compared to schema-1.

The design of the main page that the user uses to enter the document, the design of an index page that is useful for correcting whether the entered document is correct. The results page is divided into two conditions: the first condition for documents with below-grade results and the second condition for documents with above-grade results. The final project page is designed to display the entire final project document in the database.

5. Conclusions

Based on the performance test of the Kipcheck in this research, it can be concluded that: (i) based on section 4.3, the Kipcheck is capable of processing 14938 words in the input document, (ii) based on section 4.4, the comparison of results between manual calculations and calculations of Kipcheck has an accuracy of 99.88% from 100%. This shows the closeness of the results between Kipcheck and the actual cosine similarity calculation results. Calculating the similarity value between the input document and the source document displayed by Kipchek is the true value of the similarity of the documents. (iii) testing of cosine similarity consistency results between schema-1 and schema-2 shows that schema-2 has consistent similarity results and does not depend on the number of documents in the database compared to schema-1. Therefore, scheme-2 is the best scheme in the Kipcheck in this research to be implemented.

Several suggestions can be made for further development by (i) adding new features as needed, (ii) carrying out word weighting at the stage of calculating cosine similarity, (iii) performing word exclusion at the stemming stage when calculating cosine similarity, (iv) adjusting stemming library with the main language used in the document.

Bibliography

[1] I. Pahlevi and M. A. Bijaksana, "Perhitungan Kemiripan Dokumen Bahasa Indonesia Menggunakan Metode Cosine Similarity (Studi Kasus : Abstrak Tugas Akhir Fakultas Informatika Universitas Telkom)," Open Library Telkom University, Bandung, 2016.

[2] D. Gunawan, C. A. Sembiring and M. A. Budiman, "The Implementation of Cosine Similarity to Calculate Text Relevance between Two Documents," IOP, 2018.

[3] T. Foltýnek, D. Dlabolová, A. A. Naumeca, S. Razı, J. Kravjar, . L. Kamzola, J. G. Dib, Ö. Çelik and D. W. Wulff, "Testing of support tools for plagiarism detection," International Journal of Educational Technology in Higher Education, 2020.

[4] Maulid. Hariandi, Azimi. Indra, Fauzi. Amir, AND Sukarno. Parman, "Collaborative System for Friday Preacher Scheduling" IJAIT (International Journal of Applied Information Technology) [Online], Volume 3 Number 01 (27 December 2019)

[5] Hendriyanto. Robbi, AND Adolf Telnoni. Patrick, "Application for Final Project Collaboration and Management in School of Applied Science, Telkom University" IJAIT (International Journal of Applied Information Technology) [Online], Volume 2 Number 02 (10 August 2018)

[6] Plagiarism checker, Plagiarism Checker - 100% Free Online Plagiarism Detector, (2021), Online: Retrieved March 25, 2021, from https://smallseotools.com/plagiarism-checker/

[7] Plagiarism checker for educators and students, Unicheck, (2021), Online: Retrieved March 25, 2021, from https://unicheck.com/

[8] Plagiarism checker: 100% free and accurate - duplichecker, Duplichecker.com, (2021), Online: Retrieved March 25, 2021, from https://www.duplichecker.com/

[9] KBBI, Kamus Besar Bahasa Indonesia (KBBI), Online: Retrieved March 15, 2021, 2021.

[10] KBBI, Kamus Besar Bahasa Indonesia (KBBI), Online: Retrieved March 15, 2021, 2021.

[11] Republik Indonesia, Regulation of the Minister of National Education number 17 of 2010 on the prevention and countermeasures of plagiarism in universities, Jakarta, 2010.

[12] Republik Indonesia, Law No. 20 of 2003 on the national education system, Jakarta, 2003.

[13] R. P. Pratama, Aplikasi Deteksi Plagiarisme Menggunakan Metode Cosine Similarity, Malang, 2018.

[14] S. P. Gunawan, L. D. Krisnawati and A. R. Chrismanto, "Analisis Fitur Stilometri dan Strategi Segmentasi pada Sistem Deteksi Plagiasi Intrinsik Teks," Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi), pp. 988-997, 2020.

[15] R. T. Wahyuni, D. Prastiyanto and E. Supraptono, Penerapan Algoritma Cosine Similarity dan Pembobotan TF-IDF pada Sistem Klasifikasi Dokumen Skripsi, Semarang: Jurnal Teknik Elektro, 2017.

[16] N. Tahaei and D. C. Noelle, "Automated Plagiarism Detection for Computer Programming," *ICER (International Computing Education Research),* vol. 8, pp. 178-186 , 2018.