Journal of Artificial Intelligence and Engineering Applications

Website: https://ioinformatic.org/

15th October 2024. Vol. 4. No. 1; e-ISSN: 2808-4519

Application of the KARP RABIN Algorithm for Plagiarism Detection System in Thesis Proposal Submission in the Department of Informatics Engineering STMIK Kaputama

Ruth Rani Simanjuntak[1*], A M H Pardede[2], Zira Fatmaira[3]

[1, 2, 3] *STMIK KAPUTAMA*

ruthraniju@gmail.com [1*], *akimmhp@live.com* [2], *zirafatmaira0301@gmail.com* [3]

Abstract

Plagiarism is a serious threat, especially to academic honesty, so a detection system that can analyze various types of documents is needed. This research develops a plagiarism detection system using Optical Character Recognition (OCR) to convert image text into digital text. Rabin – Karp algorithm with rolling hash and Dice Coefficient Similarity is applied to measure similarities between documents. Testing is carried out on .doc, .txt, .jpg files. As a result, the system can detect plagiarism well in clear text and image documents, but accuracy can decrease in low-quality images. In conclusion, the similarity of content, sentence structure, and format affects the degree of similarity, while OCR techniques work effectively even though they are limited to low-quality images.

Keywords: *Plagiarism, OCR, Rabin-Karp, Rolling Hash*

1. Introduction

The development of artificial intelligence technology needs to remind students how to use internet access ethically in creating scientific papers. Easy access to information allows students to easily copy and imitate the work of others without giving appropriate credit. This raises concerns about academic honesty and plagiarism. The consequences obtained by plagiarism violators by plagiarizing, having their titles revoked and given sanctions such as publication cancellation, disqualification, academic sanctions, cancellation of diplomas obtained. STMIK Kaputama Binjai has established a policy against plagiarism, especially in the context of writing scientific papers and journal publications. The university uses tools such as Turnitin to check the level of plagiarism in journal articles to be published and any indication of plagiarism will result in a rejection of the article. In the process of submitting thesis proposals, there is no system to manage the percentage of plagiarism of student theses. It is feared that the shortcomings of this system can later cause several problems such as a decline in academic reputation due to a lack of awareness and education about plagiarism.

2. Literature Review

2.1. System

This system is an interconnected network of actions that come together to carry out an activity or achieve a specified goal. Another definition explains that a system is basically a set of elements that are closely related to each other, that work together to achieve a certain goal, and in a simple way we can interpret a system as a set or set of elements, components or variables that are organized, interrelated, interdependent and integrated.

2.2. Plagiarism

Definition that explains that plagiarism is taking essays, opinions, and so on from other people and making them seem as if they are their own essays/opinions, for example publishing other people's written works on their own behalf. People who plagiarize are called plagiarists or plagiarists.

2.3. Rabin-Karp Algorithm

The Rabin-Karp algorithm is one of the algorithm techniques that uses hashing techniques, namely the method used in fingerprint documents. The Rabin-Karp algorithm is one of the string search algorithms developed by Michael O. Rabin and Richard M. Karp in 1987 that uses hashing functions to find patterns within text strings. The Rabin-Karp algorithm has several characteristics, namely using k-grams and hashing. The application of the Rabin-Karp algorithm is carried out after passing the preprocessing stage.

2.4. Dice Coefficient Similarity

There are various similarity methods that can be used to analyze data such as pattern recognition, clarification and natural language processing (NLP). One of them is text-based language processing, namely Dice Coefficient Similiarity. This method is a method that can be used in calculating the similarity of texts using the K-Gram approach to calculate the similarity value with the K-Gram approach:

$$S = \frac{2\,X\,C}{A+B} \quad \text{.....(1)}$$

Description: S is the value of Similarity, A and B are the sum of the fingerprint hashes in text 1 and fingerprint hashes in text 2, C is the sum of the combined fingerprint hashes A and B. Fingerprint hashes are unique and non-duplicate hashes.

2.5. Thesis Proposal

A thesis is a scientific work that conveys the results of research, both experimental and field studies, in writing. Research is a systematic, planned, and directed process to collect facts, data, or information about a problem in a certain field of science by using scientific methods to examine or answer problems or test hypotheses. Research must follow the principles of argumentation, objectivity, openness, and independence.

2.6. Informatics Engineering

Informatics engineering is one of the majors in college that studies how computer science and mathematical analysis are applied in the design, testing, development, and evaluation of operating systems, software, and computer performance. The various branches of computer science include programming, computer networking, cybersecurity, databases, artificial intelligence, and software application development in this program. Informatics Engineering students will spend a lot of time with programming, which is basically the embodiment of algorithms (mathematics and logic).

2.7. XAMPP

Xampp is a web server-based software that is open source (free), and supports various operating systems, whether Windows, Linux, or Mac OS. Xampp is used as a standalone server or commonly referred to as localhost. This makes it easier to edit, design, and develop the application.

2.8. PHP

PHP or the abbreviation of Hypertext Preprocessor is one of the open source programming languages that is very suitable or specialized for web development and can be embedded in an HTML script. The PHP language can be said to describe several programming languages such as C, Java, and Perl and is easy to learn. PHP is a server-side scripting language, where data processing is done on the server side.

2.9. Waterfall Models

The software development model used by the author is the waterfall model. This model is also referred to as a waterfall model.

2.10. OCR

Optical Character Recognition is responsible for recognizing text in recognizing text characters in images and converting them into the American Standard Code for Information Interchange (ASCII) or other equivalent and editable machine language.

3. Analysis and Design

3.1. Research Methodology

The research methodology is a series of stages taken during the research. To achieve the objectives of the research, a methodology is needed as a guide in each step of the research carried out.

Fig. 1: Structure of Research Methodology

1. Literature Study, in this phase the researcher looks for references from previous research on thesis writing. Authors also look for relevant and legitimate sources, such as books, journals, and theses.
2. Data collection, in this phase the data will be collected into the application database. Part of the thesis chapter that will use it is chapter one.
3. System Design, in this phase involves the design of a software designer system for the purpose of monitoring the interface and hardware or tools used in the research that will be carried out on plagiarism detection using the Rabin-Karp Algorithm. This stage is carried out after pre-processing data with k-gram values that have been parsed and forming several windows. Each window is calculated a hash value and its similarity percentage value is searched using dice coefficient similarity and the software development model uses the waterfall model.
4. Conclusions and suggestions, in this phase the author summarizes and considers the main findings of the research produced. The conclusion includes details of the results and their impact on the problem or scientific topic used. In this phase, development suggestions for future research are also given.

3.2. Data Collection Methods

The data collection carried out in the study is by observing student theses that have been published by comparing thesis proposals that will be published. This research uses chapter one of the thesis.
The stages that will be carried out are, collecting thesis references for students of the Informatics Engineering study program at STMIK Kaputama university. The process that will be carried out is by converting pdf to text so that the value can be calculated with the repository contained in the database. This document will later go through a pre-process stage before testing the rabin-carp algorithm.

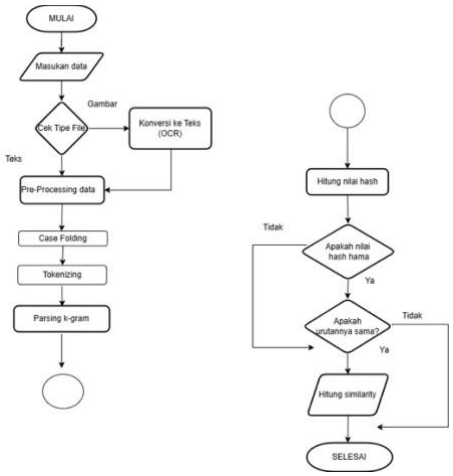

Fig. 2: Plagiarism Detection Flowchart using Rabin-Karp Algorithm

After collecting data, the steps taken to implement the Rabin-Karp algorithm are:
Example data:
Original Document (A):

Metode *Backpropagation* merupakan salah satu algoritma yang sering digunakan dalam menyelesaikan masalah-masalah yang rumit. Hal ini

Fig. 3: Sample Test Document

Thesis file "Artificial Neural Network Predicts Marriage at the Ministry of Religion of Binjai City Using the Backpropagation Method

Test Document (B): "The algorithm used to solve a difficult problem is backpropagation"

algoritma yang digunakan untuk menyelesaikan masalah yang sulit adalah backpropagation

The two examples above have different file forms. In the original document, the input provided is a picture while in the test document, the input given is text. Conversion to get the text from the image uses OCR (Optical Character Recognition) techniques. In the development of the system, OCR libraries will be added to the programming language used. After getting the text results, the next stage is pre-processing the data.

1. Pre-processing Data
   - Casefolding, the process of converting all letters in text to lowercase or uppercase. In addition to letter characters, other characters are omitted.

Original Document (A):

metode backpropagation merupakan salah satu algortima yang sering digunakan untuk menyelesaikan masalah-masalah yang rumit. Hal ini

Test Document (B):

algoritma yang digunakan untuk menyelesaikan masalah yang sulit adalah backpropagation

   - Tokenizing, the process of removing punctuation marks so that it produces smaller words and then converts them into tokens.

Original Document (A):

[“metode”, “backpropagation”, “merupakan”, “salah”, “satu”, “algortima”, “yang”, “digunakan”, “untuk”, “menyelesaikan”, “,masalah”, “masalah”, “yang”, “rumit”, “hal”, “ini”]

Test Document (B) :

[“algortima”, “yang”, “digunakan”, “untuk”, “menyelesaikan”, “masalah”, “yang”, “sulit”, “adalah”, “backpropagation”]

2. Rabin-Karp Algorithm
After going through pre-processing, the Rabin-Karp algorithm will be applied by determining the K-Gram value and hashing.
After the two are calculated, the similarity percentage will also be determined using the dice coefficient similarity. The process is explained as follows:

3. K-gram, cutting letter characters from the beginning of the text to the end of the text. The number of k-grams used is the value of k=5.

Original Document (A):

[“metod", “etode”, “todeb”, “odeba”, “debac”,”eback”,”backp”,, “ackpro”, “ckpro”, “kprop”, “ropag”, “opaga”, “pagat”, “agati”, “gatio”, “ation”, “tionm”, “onmer”, “nmeru”, “merup”, “rupak”, “upaka”, “pakans”, “akans”, “kansa”, “ansal”, “nsala”, “salah”, “alahs”, “lahsa”, “ahsat”, “hsatu”, “satua”, “atuag”, “tuago”, “ualgo”, "algor", "lgori", "gorit", "oritm", "ritma", “timay”, “imaya”, “mayas”, “ayang”, “yangs”, “angs”, “ngser”, “gseri”, “serin”, “ering”, “ringd”, “ingdi”, “ndig”, “gdigu”, “digun”, “iguna”, “gunak”, “unaka”, “nakun”, “akunt”, “kuntu”, “untuk”, “ntuks”, “tukme”, “ukmen”, “kmeny”, “menye”, “enyel”, “nyels”, “yeles”, “elesa”, “lesai”, “esaik”, “saika”, “aikan”, “ikanm”, “kanma”, “masal”, “asalm”, “salma”, “almal”, “lmalh”, “malha”, “halni”, “alini”]

Test Document (B):

[“algor”, “lgori”, “gorit”, “oritm”, “ritma”, “itmay”, “tmayg”, “mayan”, “ayang”, “yangd”, “angdi”, “ngdig”, “gdigu”, “digun”, “iguna”, “gunak”, “unaka”, “nakun”, “akunt”, “kuntu”, “untuk”, “ntuks”, “tukme”, “ukmen”, “kmeny”, “menye”, “enyel”, “nyele”, “yeles”, “elesa”, “lesai”, “esaik”, “saika”, “aikan”, “ikanm”, “kanma”, “anmas”, “nmasa”, “masal”, “asala”, “salah”, “alahy”, “lahya”, “ahyan”, “hyang”, “yangs”, “angs”, “ngsul”, “gsuli”, “sulit”, “ulita”, “litad”, “litad”, “itada”, “tadal”, “adala”, “dalah”, “alahb”, “lahba”, “ahbac”, “hback”, “backp”, “ackpr”, “ckpro”, “kprop”, “propa”, “ropag”, “opaga”, “pagat”, “agati”, “gatio”, “ation”

4. Hashing, the process of converting string values into integers to generate a unique hash value for each substring in the text.

$$H(c1 \dots\dots ck) = (c1 * b^{k-1} + c2 * b^{k-2} + c3 * b^{k-3} + \cdots + + cn * b^{k-n}) \, mod \, q \qquad \textbf{(1)}$$

Example words:
Original document (A) with the word "algor", with the value p=31 and m=101. The ASCII of "algor" is a (97), l (108), g (103), o (111), r (114).

$$H(c1 \dots\dots ck) = (c1 * b^{k-1} + c2 * b^{k-2} + c3 * b^{k-3} + \cdots + + cn * b^{k-n}) \, mod \, q$$

$$H = (97\, x31^4 + 108\, x31^3 + 103\, x31^2 + 111\, x31^1 + 114\, x31^0 \, mod \, 101$$

$$H = (97\, x\, 923521 + 108\, x\, 29791 + 103\, x961 + 111\, x\, 31 + 114\, x\, 1\,) \, mod \, 101$$

$$H = (89659717 + 3262178 + 99123 + 3441 + 114\,) \, mod \, 101 \; H = 92957073 \, mod \, 101 = 87$$

So, the value of the string "algor" with (b=31) and (q=101) is 87.
Each word that has been formed through k-grams is calculated using the hash function so that it produces a new value that will later be compared with other sentences.
The following is the hash value of the original document and the hash of the test document with the value b=31, q=101, k-gram = 5:

Original Documents:

[97, 24, 65, 81, 44, 51, 77, 59, 98, 7 , 47, 10, 81, 61, 42, 76, 54, 42, 79, 62, 79, 90, 6, 91, 67, 61, 52, 69, 61, 45, 38, 64, 37, 93, 47, 57, 35, 61, 84, 87, 51, 22, 96, 93,52, 82, 55, 85, 100, 20, 21, 2, 83,33, 12, 50,61, 42, 10, 98, 38, 95, 20, 83, 40, 70, 58, 50, 33, 28, 96, 1, 45, 19, 99, 43, 68, 84, 71, 85, 59, 84, 39, 53, 11, 60,21, 83, 89, 50]

Test Documents:

[87, 51, 22, 96, 2, 59, 37, 50, 85, 85, 32, 61, 42, 10, 98, 38, 75, 20, 83, 40, 70, 58, 50, 33, 28, 96, 1, 31, 19, 99, 43, 68, 84, 71, 64, 85, 59, 84, 39, 41, 38, 70, 21, 96, 25, 100, 20, 6, 42, 16, 71, 74, 9, 7, 0, 80, 47, 15, 0, 83, 77, 59, 98, 7, 47, 10, 81, 61, 42, 76, 54 {78} {72} {98} {22} {13} {79} {39} {41} {38} {16} {77} {59} {98} {7} {47} {10} {81} {61} {42} {76} {54}]

The total hash value of each document:
Test Documents = 91
Original Documents = 71
After getting the hash value of both documents, then a rolling hash is applied to find the same hash value.
Same hash:
{96} {2} {59} {37} {50} {85} {61} {42} {10} {98} {38} {20} {83} {33} {28} {1} {19} {99} {43} {68} {84} {71} {39} {61} {45}
The same number of hashes of the original document and the test document is

$$\sum Hash\ asli\ \cap Hash\ uji\ = 25$$

5. Dice coefficient similarity, a metric used to measure similarity or similarity between two documents or series of words.

$$S = \frac{2\ X\ C}{A+B} \text{.....(2)}$$

Same k-gram count: 25

Dice Coefficient Similarity (S) $= \frac{2\ X\ C}{A+B} = \frac{2\ X\ 25}{91+71} = \frac{52}{162} =$ 0,308

So, the percentage of similarity between the two sentences is about 0.308 or about 30.8%. This shows that the two sentences have similarities that fall into the category of "medium similarity".

4. Implementation and testing

In this chapter, the implementation of the Rabin-Karp algorithm in designing a plagiarism detection system is discussed. The things that will be discussed are the implementation of device specifications, interfaces, function implementation, and testing of the system using the blackbox method.

4.1. User Interface Implementation

Based on the design that has been carried out in the previous chapter, then it is implemented in a system. On the main page display there are several menus such as Home, Check Plagiarism, Test History, Library, and Admin. The appearance can be seen in the image below.

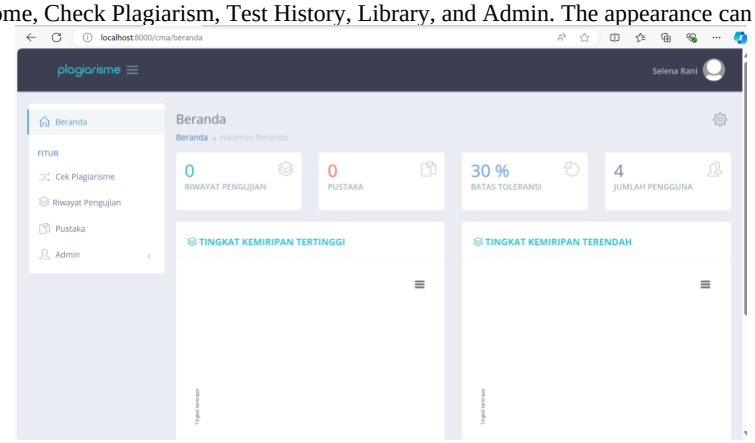

Fig. 4: Homepage Form

Next is the Plagiarism Check view to compare two documents. In the system that is built, the document materials to be tested can be in the form of document files with .txt, .pdf, .jpg, .png extensions. The K-Gram value is already determined on the programming page.

Fig. 5: Testing Form 1

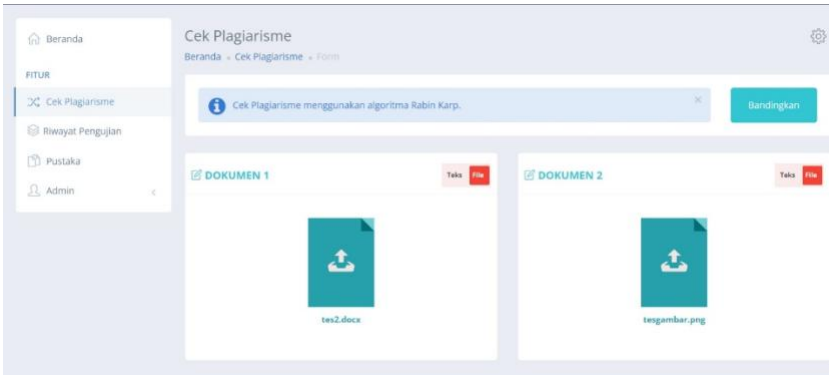

Fig. 6: Testing Form 2

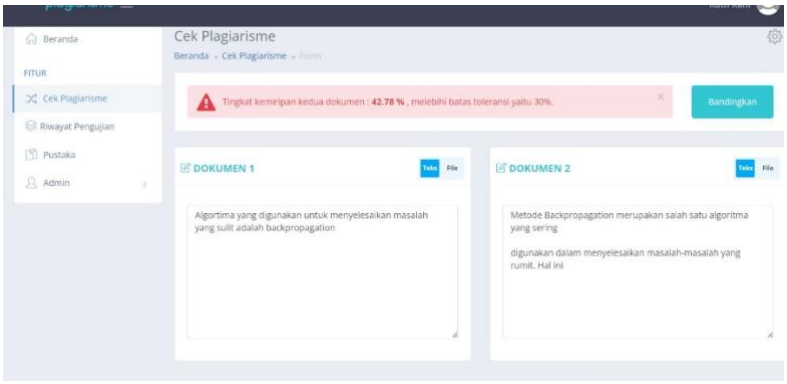

Fig. 7: Testing Form 3

Furthermore, on the Test History Page, several document files that have been uploaded previously will be displayed. The display is in the picture below.

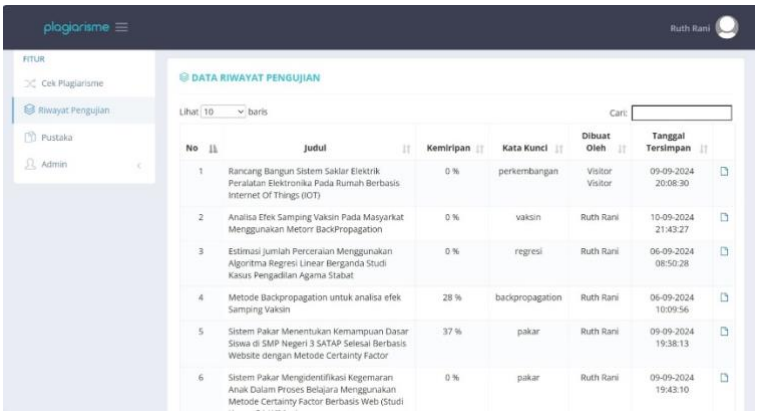

Fig. 8: Library History Form

Furthermore, on the Library page, the admin can save references to other documents that will be tested with other documents. The page display is in the image below.

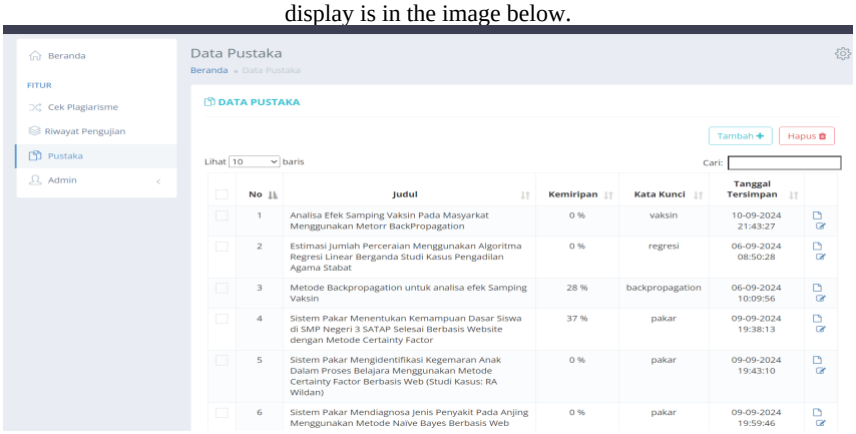

Fig. 9: Library Form

Furthermore, on the Admin Page, there are several options to set plagiarism tolerance limits, users, and other settings. The display is in the picture below.

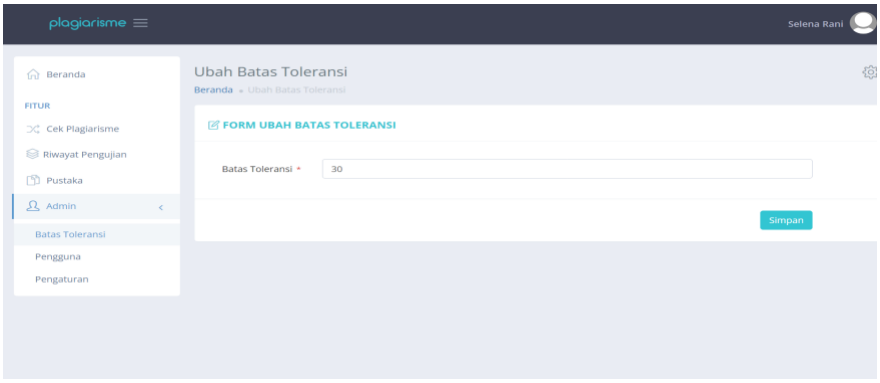

Fig. 10: Form Admin

4.2. Testing

The system testing carried out on this system uses black box testing. Black-box testing is to input and test whether the functions contained in a running system are fit for purpose, and are reflected in the output. Here are the results of the test:

Original document - T1:

Penerapan vaksinasi pada masyarakat telah menjadi bagian penting dalam mencegah penyebaran penyakit menular. Vaksin membantu tubuh dalam membentuk antibodi terhadap patogen, sehingga mampu mencegah infeksi. Namun, ada sejumlah kecil kasus efek samping yang dilaporkan, meskipun sebagian besar bersifat ringan seperti demam atau nyeri di tempat suntikan.

Test Document – T2 :

Penerapan vaksinasi di masyarakat telah menjadi elemen penting dalam pencegahan penyebaran penyakit infeksius. Vaksin membantu tubuh membentuk antibodi untuk melawan patogen tertentu, sehingga mampu mencegah infeksi. Namun, ada sejumlah kecil laporan mengenai efek samping yang sebagian besar bersifat ringan seperti demam atau nyeri di tempat suntikan.

Original Documents – K1 :

Menurut penelitian yang dipublikasikan oleh Smith et al. (2021), "Penerapan vaksin COVID-19 secara global telah memberikan dampak signifikan terhadap penurunan tingkat kematian dan infeksi. Namun, masih terdapat kekhawatiran mengenai efek samping jangka panjang yang belum sepenuhnya diketahui." Penelitian ini menyoroti pentingnya pengawasan berkelanjutan untuk memastikan keamanan vaksin dalam jangka panjang.

Test Document – K2:

Berdasarkan penelitian yang diterbitkan oleh Smith et al. (2021), "Distribusi vaksin COVID-19 secara luas telah menunjukkan dampak besar terhadap penurunan angka kematian dan kasus infeksi. Namun, kekhawatiran tentang efek samping jangka panjang yang belum sepenuhnya dipahami masih ada." Studi ini menekankan pentingnya pengawasan terus-menerus untuk menjamin keamanan vaksin dalam jangka waktu lama.

Original Document – G1:

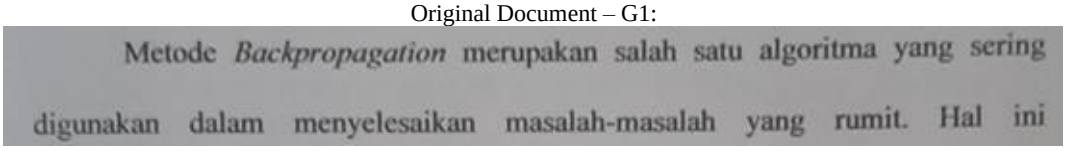

Fig. 11: Original Document – G1

Test Document – G2:

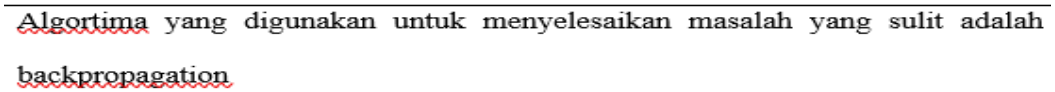

Fig. 12: Test Document – G2

T(1) and T(2) are files with the extension .txt or .docx, K(1) and (K2) are texts that have citations from previous research. These two documents, both text and citations, have the same and similar context. G(1) is a file with a jpg extension. G(1) and (G2) have different contexts than T and K files.
Each document is compared to each other for similarity testing. The test results are in the table below.

Table 1: Document Testing

No	Dokumen	Similarity	Konteks Kesamaan
1	T1; T2	71.02%	√
2	K1; K2	55,80%	√
3	G1; G2	47,06%	√
4	T1; K1	15,19%	√
5	T1; K2	15,19%	√
6	T1; G1	3,90%	X
7	T1; G2	1,61%	X
8	T2; K2	13,90%	X
9	T2; G2	2,67%	X
10	K2; G2	1,66%	X

Based on the comparative test data of the above documents, namely:

1. The highest level of similarity is available for T1 and T2 testing, which is 71.02%. This shows that the two text files have a lot in common, both in terms of content and sentence structure.
2. Similarity of Citations (K1 and K2) has a fairly high similarity of 55.80%. This can be interpreted that the two quotes have the same substance or topic, but there are still differences in the details or the way they are written.
3. Low Similarity in T1 and K1 Document Combinations. This shows that the text in the citation document differs significantly compared to the text document tested.
4. The comparison between text and images (T1 and G1) shows very low similarity. This is because the two document contexts are different, and the OCR results can detect image files and convert them to text. These differences are due to writing styles, or even minor errors in OCR results.

From the results of this test, it can be concluded that the plagiarism detection method using the Rabin-Karp algorithm is effective in detecting similarities between text documents. However, the accuracy of the results may decrease when comparing text originating from OCR. This can be affected by image quality, image distortion, or limitations of the algorithm.

5. Conclusion

Based on the discussion in the previous chapters, in this chapter the author can draw conclusions including:

1. This application can be used to detect plagiarism well using the Rabin Karp and Dice Coefficient Similarity algorithms. Files that have similarities in terms of content, sentence structure, topic, and writing format can affect the level of similarity to be higher.
2. The OCR method can be used to help the system convert images into text files so that the results can be used to calculate the hash value.

Reference

[1] Erawati, W. (2019). Designing a Sales Information System with a Waterfall Method Approach. Journal of Media Informatics Budidarma, 3(1), 1. https://doi.org/10.30865/mib.v3i1.987

[2] Dermawan, M. S., Mulyawan, B., & Lauro, M. D. (2019). Designing a document management system application and text search using Optical Character Recognition (OCR). Journal of Computer Science and Information Systems, 7(1), 81–86.

[3] Filcha, A., & Hayaty, M. (2019). Implementation of the Rabin-Karp Algorithm for Plagiarism Detection in Student Assignment Documents. JUITA : Journal of Informatics, 7(1), 25. https://doi.org/10.30595/juita.v7i1.4063

[4] Ginting, S. L. B., Ginting, Y. R., Sutono, S., & Sirait, W. A. (2022). The word similarity detection application uses the Rabin-Karp algorithm. Journal of Technology and Information, 12(2), 162–175. https://doi.org/10.34010/jati.v12i2.6947

[5] Hartanto, A. D., Syaputra, A., & Pristyanto, Y. (2019). Best parameter selection of rabin-Karp algorithm in detecting document similarity. 2019 International Conference on Information and Communications Technology, ICOIACT 2019, 457–461.

[6] W., Utami, E., & Sunyoto, A. (2022). Selection of the Best K-Gram Value on Modified Rabin-Karp Algorithm. IJCCS (Indonesian Journal of Computing and Cybernetics Systems), 16(1), 11.

[7] Yaqin, A., Dahlan, A., & Hermawan, R. D. (2019). Implementation of algorithm rabin-karp for thematic determination of thesis. 2019 4th International Conference on Information Technology, Information Systems and Electrical Engineering, ICITISEE 2019, 395–400.

[8] Pratiwi, M. A., & Aisya, N. (2021). The phenomenon of academic plagiarism in the digital age. Publishing Letters, 1(2), 16–33.