Bit-Tech (Binary Digital - Technology)
Vol.6, No.2, December 2023
Available online at: http://jurnal.kdi.or.id/index.php/bt

The Use of Information Retrieval in Student Academic Document Plagiarism Detection System

Arief Rahman Yusuf[1)*], Angga Prasetyo[2)]

[1)2)]*Universitas Muhammadiyah Ponorogo*
Jalan Budi Utomo No. 10, Ponorogo, Indonesia
[1)]yusuf@umpo.ac.id
[2)]angga_raspi@umpo.ac.id

Article history:

Received 14 Dec 2023;
Revised 16 Dec 2023;
Accepted 19 Dec 2023;
Available online 28 Dec 2023

Keywords:

Academic Documents
Detection System
Information Retrieval
Plagiarism
Turnitin

Abstract

The practice of plagiarism can threaten the credibility of academic documents such as final assignments, theses, theses, and dissertations. Universitas Muhammadiyah Ponorogo (UMPO) is one of the best Muhammadiyah Universities in Indonesia. All academic documents at UMPO are well stored. However, plagiarism is only found in the Turnitin database and in the title of academic documents. This research aims to pre-screen academic documents in the internal scope by creating a plagiarism application that is easy to use and cheaper in terms of cost than using Turnitin. The steps of making a plagiarism detection system in academic documents using information retrieval (IR) with waterfall development. The design process starts from system requirements analysis, design, implementation, and testing. This plagiarism detection system goes through several stages, namely tokenisation, stopword removal, stemming, and termweigthing. Tokenisation involves converting uppercase letters into lowercase letters and removing punctuation marks. Stopword removal removes noise, stemming converts words into basic forms, and termweigthing uses local and global weighting to calculate similarity with the Vector Space Model. Document similarity is calculated using cosine similarity, rejected documents are considered free of plagiarism. If there is no similarity, the document is accepted as a source document, while the rejected document is not stored in the database. Further research on the document plagiarism detection system that previously only used the txt file extension, in the future it can use the .doc or HTML extension so as to increase the effectiveness of the performance of academic document examination time.

I. INTRODUCTION

Plagiarism is academic cheating that involves using another's language, ideas, or expressions as one's own in academic or professional work. This plagiarism can occur in different levels of work and the impact varies, ranging from possible reduction in employment opportunities to uncertainty. Therefore, it is important to understand and identify the extent of plagiarism in scientific and academic work [1]. Document plagiarism is the act of taking and using someone else's writing or work without acknowledging the original source. Plagiarism is defined as taking someone else's writing or thoughts and making them as if they were one's own [2].

Plagiarism is a form of unethical behaviour in the academic world. Plagiarism occurs when a person intentionally or unintentionally tries to gain recognition or value from a scientific work by quoting part or all of another person's scientific work without providing proper and adequate sources [3]. This action is a violation of research ethics and can be detrimental to the party who created the scientific work. Therefore, it is important for every researcher or student to avoid plagiarism by citing sources accurately and giving appropriate recognition to the original author [4].

Some plagiarism detection systems, such as Turnitin, help identify plagiarism cases more efficiently and effectively. However, there is a consensus in the literature that plagiarism detection systems are not able to identify all types of plagiarism, such as simple plagiarism and random deepened plagiarism. Therefore, it is important to continuously develop and improve plagiarism detection systems to achieve a higher level of accuracy. In this context, some commonly identified types of plagiarism include: Simple plagiarism: Involves the copying of text

* Corresponding author

ISSN 2622-2728 (online) 2622-271X (print)
doi: 10.32877/bt.v6i2.1063

as speech, which may involve the addition, substitution, or alteration of the original text. Random deepened plagiarism: Involves the use of original text with random characters or different characters, which can be done using text circulation technology or the use of altered text. Translated plagiarism: Involves the use of the original text in a language different from the original language. Idea plagiarism: Involves the use of ideas or concepts from another source without giving due credit to the original author [5].

Plagiarism detection is very important in the academic world, especially in universities because it can threaten the credibility of academic documents such as final assignments, theses, theses, and dissertations. Universitas Muhammadiyah Ponorogo (UMPO) is one of the best muhammadiyah universities in East Java. All academic documents at UMPO are well kept. Since the number of academic documents of UMPO students is in the thousands, software is needed to identify this early stage of plagiarism. Software to detect plagiarism of student academic documents at UMPO is needed due to the huge number of documents. In addition, to subscribe to Turnitin software requires a lot of money. The purpose of making this software is to find back the stored data and then provide information about the subject needed. The term for this software is information retrieval (IR).

Modern IR systems have evolved to cover a wide range of data types and information sources. IR is concerned with automatically managing information from large data collections to retrieve relevant and useful information based on user requests [6]. Recent developments include multimedia and spatial data sources and the application of AI. Much of the research on IR is the application of information retrieval to web search engines and its evolution to provide relevant information and continuously adapt to user needs [7]. In addition, research on the application of cosine similarity algorithm and matrix method in information retrieval to find relevant approaches in retrieving online news documents [8]. The development of a digital library system to retrieve structured data stored in a database, then provide relevant information to meet user needs.[9]

II. METHODS

This plagiarism detection system uses a waterfall development model so that the design is carried out in stages, starting from the system requirements stage then moving to the analysis stage focused on the required functions and user interface, the design stage is described by DFD (Data Flow Diagram). DFD is often used to describe an existing system or a new system that will be developed with a structured, clear system and documentation of a good system [10], The implementation stage is the representation of the design into a programming language that is understood by the computer and the implementation results produced by the software, and the testing stage is designing tests in the form of input source documents, documents to be checked (copy documents), and predetermined value limits (threshold).

The design of the academic document plagiarism detection system consists of input, namely by entering documents that are considered plagiarism, the output of this system is a report on the level of similarity or similarity of an academic document in statistical form, and the process or mechanism with tokenization, namely reading documents that have .tx format, converting large alphabets into small ones, and removing punctuation marks. Stopwords removal is removing words that are included in stopwords. Stemming is converting words into basic words. Termweighting is calculating the weight of the checked document with the source document.

The system for identifying plagiarism of student academic documents examines each document in the corpus. The inputs of the academic document plagiarism detection system are as follows: a) Source document (reference document); b) Duplicate document (document to be compared); and c) Value limit (50% threshold limit).

III. RESULTS

A. Data Flow Diagram (DFD)

The process of developing a plagiarism detection system involves function and user interface analysis, as well as process analysis using a Data Flow Diagram (DFD). The DFD level 0 image is shown in Figure 1.

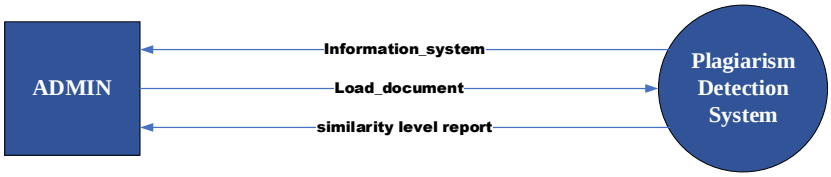

Fig. 1 DFD Level 0 Plagiarism detection system

In DFD level 0, the admin enters a copy of the document, which is then analysed for plagiarism. The input made by the admin is to enter a document that is considered plagiarism through the load document copy that has been provided by the system. The output of the system received by the admin is in the form of information on the similarity level report. From DFD level 0, each process can be described in detail into DFD level 1. DFD level 1 the DFD level of the plagiarism detection system as shown in Figure 2.

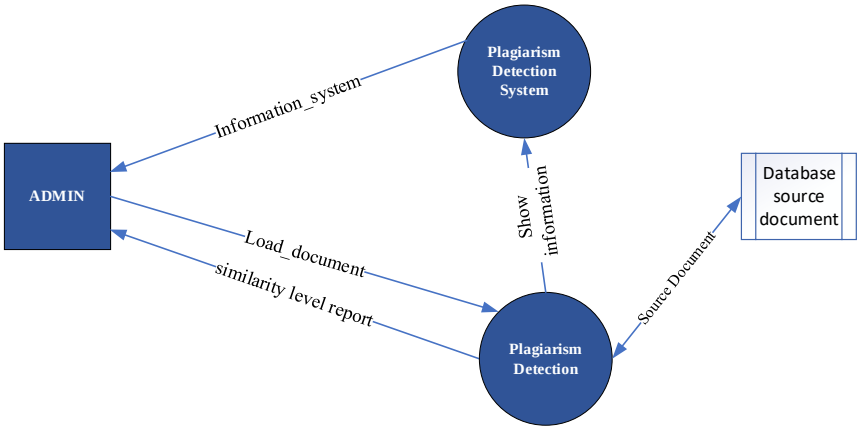

Fig. 2 DFD Level 1 Plagiarism detection system

DFD level 1 the admin performs menu select input to the system. The input made by the admin is to enter a document that is considered plagiarism through the load document copy provided by the system, after entering a document that is considered plagiarism, it will be processed. The system provides output in the form of system information and similarity level reports.

B. Entity Relationship Diagram (ERD)

ERD is a relational database modelling based on the perception that in the real world, this world always consists of a set of objects that are interconnected with each other[11]. The ERD of the plagiarism detection system is shown in Figure 3.

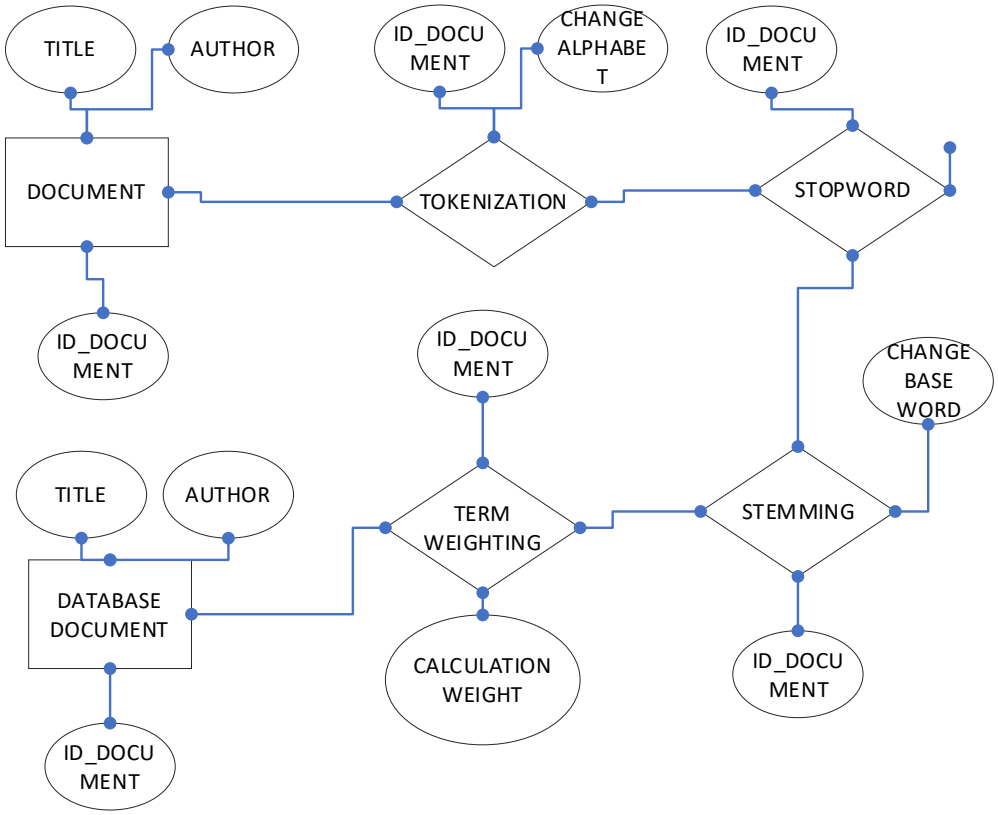

Fig 3. ERD Plagiarism detection system

The document record has a one to one relationship with the document database record because the calculation weight has one relationship in the termweighting process.

C. Flowchart

The proposed DFD system can be represented in a system flow chart, which is a tool that shows the overall performance of the system [12]. Flowchart of the plagiarism detection system as in Figure 4.

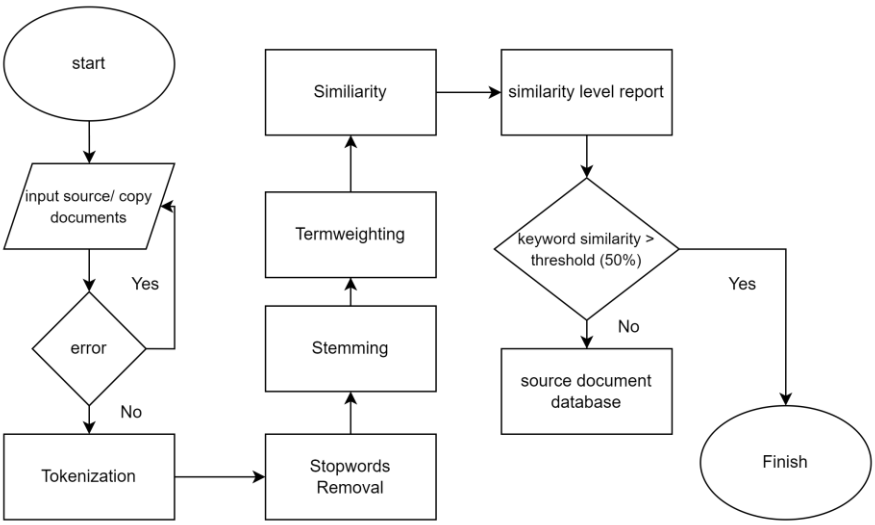

Fig. 4 Flowchart of plagiarism detection system

User interface is the part that is directly related to the user [13]. The following is the user interface of the plagiarism detection system

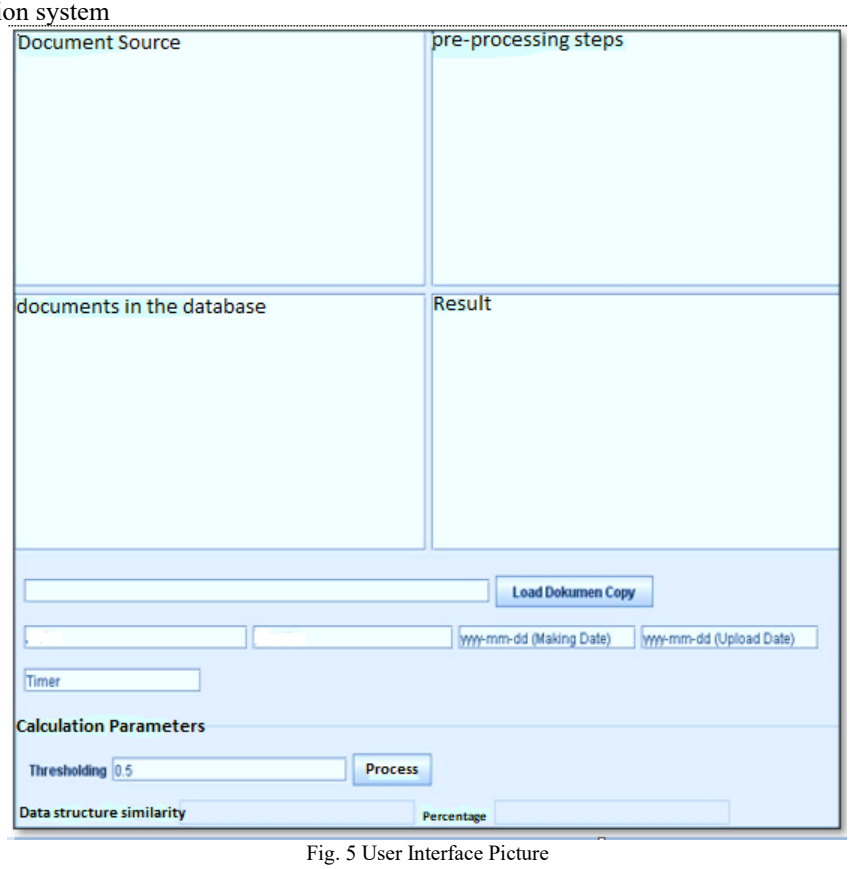

Fig. 5 User Interface Picture

The software that detects plagiarism starts with the main feature of the system, load copy documents. The identity area is the next step after load copy documents. It is used to enter the identity after filling in the identity in the identity area during the process by pressing the process button. Calculating the duration of the checking process in units of milliseconds, a timer is used. The source document area in the database, preprocessing steps, existing documents in the database, and the result of similar documents The similarity of the word similarity structure will be compared with the threshold (50%) and the statistical similarity report (percentage).

D. Test Results

Conditioning the database in an empty state is done to ensure that in the first test, the document is considered free of plagiarism because there is no comparison document in the database and it is automatically used as a source file. The document to be checked has the extension.txt and is the main target of testing the system that detects plagiarism. After the database and documents have been prepared, the next step is to perform checks to ensure that there is no plagiarism in the document as a whole. Preprocessing is part of the checking process, which consists of tokenisation, stopwords removal, stemming, and termweighting. After the preprocessing results are obtained, the similarity level of the documents is calculated using a vector space model. This calculation is done in two ways, systemically and manually.

1. Documents when the database is empty
   The first checked document is considered as the first source document in the database and the manual calculation result is 0/0 or there is no word similarity between the document and the system calculation result. The document checking time is 300.0 milliseconds.
2. Documents with 0% similarity rate Source document
   In this situation, documents that are assumed to have 0% plagiarism are not included in the database as source documents. As a result, there are two source documents stored in the system database: the document that is checked first when the database is empty and the document that is assumed to have 0% similarity rate. The process of checking the document assumed to be 0% takes about 300.0 milliseconds.
3. Documents with 75% similarity to the source document
   In the manual similarity calculation on the documents assumed to be 75%, it was found that the similarity of the documents reached 87%. However, the system calculation results show a lower figure of 65%. In this context, the document is considered plagiarised from an existing document in the database due to the detected similarity. As a consequence, plagiarised documents will be rejected by the system and will not be stored in the database as source documents. The document checking time is 70.0 milliseconds.
4. 4. Documents with 100% similarity rate Source document
   In the manual calculation, the document that is assumed to be 100% has a calculation result that reaches 100%, while the calculation system produces a figure of 98%. Therefore, the document is considered plagiarised from the document in the database because there is a similarity in the calculation results. Documents detected as plagiarised from documents in the database will be rejected by the system, so they will not be stored in the database as source documents. The time required to check a document that is assumed to be 100% is about 90.0 milliseconds.

E. Analysis of Test Results

The analysis of the plagiarism detection system reveals that the first data is considered as the source of the problem, and the subsequent data is treated as the source. This data is processed through the processes of tokenisation, stopwords removal, stemming, and term weighting. The IR process, which includes tokenisation, stopwords removal, stemming, and term weighting, is used to optimise the system. Plagiarism detection is determined by the IR threshold determined by the IR threshold [14]. From the calculated results of the four tests, an average error of 6% was obtained. One of the drawbacks of weighting using a simple and fast vector space model is that it provides irrelevant documents and may be lost due to using different words to describe the same interest [15].

IV. CONCLUSIONS

Every document that enters the plagiarism detection system must go through an IR process, including tokenisation, stemming, stopwords removal, and termweigthing. The system uses local and global tagging (tf) and idf (idf) for all documents. The results are analysed in statistical form, which shows that if there is a difference between the plagiarism detection system and the data source, the system will treat the document as a plagiarism detection system.

If there is a similarity between the checked document and the document stored in the database (which is considered plagiarism-free), the document is rejected by the system and not stored in the database. Conversely, if the calculation result of the checked document does not show any similarity with the source document, the document is stored in the database.

The suggestion for this research is that the document to be checked must pay attention to its file extension, which is txt file. In future research, the plagiarism detection system can check documents with.doc and HTML extensions. This increases time efficiency and ensures that plagiarism checking on academic documents remains effective. There is no need to use a computer with high specifications as the product specifications can be reduced.

REFERENCES

[1] E. Bensal, E. Miraflores, and N. C. Tan, "Plagiarism: Shall we turn to Turnitin?," *CALL-EJ*, vol. 14, pp. 2–22, Jan. 2013.

[2] A. R. Fadilla, H. Haryadi, and M. Rapik, "Plagiarisme Karya Ilmiah Dalam Kacamata Hukum Pidana," *PAMPAS J. Crim. Law*, vol. 4, no. 1, Art. no. 1, Feb. 2023, doi: 10.22437/pampas.v4i1.24074.

[3] G. C. Adiyati and A. Supriyanto, "PENYEBAB DAN DAMPAK BAGI SESEORANG YANG MELAKUKAN TINDAKAN PLAGIARISME DALAM PENULISAN KARYA ILMIAH," *Semin. Nas. Arah Manaj. Sekol. Pada Masa Dan Pasca Pandemi Covid-19*, no. 0, Art. no. 0, 2020, Accessed: Dec. 14, 2023. [Online]. Available: http://conference.um.ac.id/index.php/apfip/article/view/375

[4] S. Wachidah, "PLAGIARISME DALAM KATA-KATA MAHASISWA: ANALISIS TEKS DENGAN PENDEKATAN FUNGSIONAL," *Linguist. Indones.*, vol. 31, no. 2, Art. no. 2, Aug. 2013, doi: 10.26499/li.v31i2.8.

[5] T. Foltýnek, N. Meuschke, and B. Gipp, "Academic Plagiarism Detection: A Systematic Literature Review," *ACM Comput. Surv.*, vol. 52, no. 6, p. 112:1-112:42, Oct. 2019, doi: 10.1145/3345317.

[6] L. Wulandari, "PENERAPAN TEXT MINING PADA SEARCH ENGINE (STUDI KASUS E-COMMERCE SHOPEE)," *J. Teknol. Inf. Manaj. Dan Bisnis Digit.*, pp. 21–27, Sep. 2023.

[7] N. Ahmad, A. A. Prasetyo, and A. Masruri, "PENERAPAN INFORMATION RETRIEVAL PADA SEARCH ENGINE," *Knowl. J. Inov. Has. Penelit. Dan Pengemb.*, vol. 1, no. 1, Art. no. 1, Dec. 2021.

[8] F. Wiranto and I. M. Tirta, "Information Retrieval Using Matrix Methods," presented at the International Conference on Mathematics, Geometry, Statistics, and Computation (IC-MaGeStiC 2021), Atlantis Press, Feb. 2022, pp. 167–172. doi: 10.2991/acsr.k.220202.032.

[9] E. Fitriani, R. E. Indrajit, and R. Aryanti, "Penerapan Model Information Retrieval Untuk Pencarian Konten Pada Perpustakaan Digital," *J. Perspekt.*, vol. 15, no. 2, Art. no. 2, Sep. 2017, doi: 10.31294/jp.v15i2.2350.

[10] A. Sutanti, M. K. Mz, M. Mustika, and P. Damayanti, "RANCANG BANGUN APLIKASI PERPUSTAKAAN KELILING MENGGUNAKAN PENDEKATAN TERSTRUKTUR," *Komputa J. Ilm. Komput. Dan Inform.*, vol. 9, no. 1, pp. 1–8, Mar. 2020, doi: 10.34010/komputa.v9i1.3718.

[11] R. Nurmasari, S. Pinem, and U. Nurkhalifah, "Perancangan Pengelolaan Data Pelabuhan Perikanan Nusantara (PPN) Pelabuhan Ratu Menggunakan Entity Relationship Diagram (ERD)," *J. Ilm. Rekayasa Dan Manaj. Sist. Inf.*, vol. 9, no. 1, Art. no. 1, Mar. 2023, doi: 10.24014/rmsi.v9i1.22024.

[12] Z. Tuasamu *et al.*, "Analisis Sistem Informasi Akuntansi Siklus Pendapatan Menggunakan DFD dan Flowchart Pada Bisnis Porobico," *J. Bisnis Dan Manaj. JURBISMAN*, vol. 1, no. 2, Art. no. 2, May 2023, doi: 10.61930/jurbisman.v1i2.181.

[13] N. Normah and F. Sihaloho, "Perancangan User Interface (UI) dan User Experince (UX) Aplikasi pendistribution alat-alat kesehatan pada perusahaan PT. Rekamileniumindo Selaras Jakarta Barat," *Indones. J. Softw. Eng. IJSE*, vol. 9, no. 1, Art. no. 1, Jun. 2023, doi: 10.31294/ijse.v9i1.15467.

[14] L. D. Krisnawati, J. F. Lim, and G. Virginia, "Penggunaan Pemodelan Topik dalam Sistem Temu Kembali Dokumen Termirip," *J. Linguist. Komputasional*, vol. 6, no. 1, Art. no. 1, Apr. 2023, doi: 10.26418/jlk.v6i1.78.

[15] R. Kaban, P. Sihombing, M. Pandia, and P. Simamora, "Pemrosesan Query dan Pemeringkatan Hasil dalam Information Retrieval: Sebuah Kajian Literatur," *J. Inf. Syst. Res. JOSH*, vol. 4, no. 3, Art. no. 3, Apr. 2023, doi: 10.47065/josh.v4i3.2867.