PLAGIARISM DETECTION SYSTEM USING THE LEVENSHTEIN DISTANCE METHOD FOR THE BALINESE LANGUAGE

Ida Bagus Ary Indra Iswara [1)], I Made Gede Oka Graha Adinata [2)], Ida Bagus Gede Sarasvananda [3)], I Gusti Made Ngurah Desnanjaya [4)]

[1)] Institut Bisnis dan Teknologi Indonesia
indraiswara@instiki.ac.id
[2)] Institut Bisnis dan Teknologi Indonesia
okagraha51@gmail.com
[3)] Institut Bisnis dan Teknologi Indonesia
sarasvananda@instiki.ac.id
[4)] Institut Bisnis dan Teknologi Indonesia
desnanjaya@instiki.ac.id

ABSTRACT

Balinese language is one of the 718 regional languages in Indonesia. Balinese language is currently the language that is often and always used by Balinese people living in Bali and outside Bali. As time goes by, the development of the Balinese language, especially Balinese literary works, is increasing. This increase in Balinese literary works also raises several problems, one of which is the existence of plagiarism in Balinese literary works. With these problems, this research raises how to detect similarities in Balinese literary works, because it is important for us to know this so that Balinese literary works can be protected and look original. This study discusses how the performance of the Levenshtein Distance algorithm if it is implemented in Balinese literature. This study observes the performance of the Levenshtein Distance Algorithm on the computer CPU. The results of the observations show that if the document has the same number of words with different contents and is repeated 5 times in each test, the system gets a similarity value of 23% and the highest is 34%, where the average CPU usage is 33%-34%. If compared with the manual calculation of the Levenshtein Distance Algorithm, the results of the system with manual calculations are the same.

Keywords: *Plagiarism, Balinese Language, Levenshtein Distance, Similarities, Documents*

1. INTRODUCTION

Plagiarism is the act of taking someone's work without citing the original source. According to Habibi & Munir (2009) plagiarism is the act of submitting or presenting other people's ideas or words/sentences without citing the source. Plagiarism causes the work that has been taken by the perpetrator to be of no value to the original author because the work is taken and recognized as the property of the perpetrator (Febiawan et al., 2019).

Document plagiarism can be detected by comparing two documents or by using a plagiarism checker application that is available on the internet. Plagiarism checkers available on the internet generally only detect documents with specific languages such as English and Indonesian, but Balinese language documents cannot be detected with this specific plagiarism checker because Balinese language has a different morphology from other languages, therefore documents Balinese language cannot be detected by the plagiarism checker which is currently available (Febiawan et al., 2019; Hutagalung, 2019). Over time, the development of the Balinese language, especially Balinese literary works, is increasing. This increase in Balinese literary works also raises several problems, one of which is the existence of plagiarism in Balinese literary works which can harm Balinese literary activists.

Seeing the importance of Balinese literary works in order to continue to exist, the researcher considers that the detection system for Balinese literature needs to be developed, because the existence of a similarity detection system for literary works, especially those in Balinese language, will make Balinese literary activists more enthusiastic in spawning their works.

This study uses the Levenshtein Distance method which has been proven to be good in detecting similarities in Indonesian documents based on research (Sari et al., 2021). The Levenshtein Distance method was created by Vladimir Levenshtein in 1965. The Levenshtein Distance method uses the edit distance calculation obtained from the matrix to calculate the number of string differences between two strings. The calculation of the distance between these two strings is determined from the minimum number of change operations to make string A into string B. There are 3 main operations that can be performed by this algorithm, namely, insertion, deletion, and substitution (Rustamovna, 2021).

2. RESEARCH METHODOLOGY

Our research is based on a five-stage design science research methodology (DSRM) (Peffers et al., 2007): 1. Identification; 2. Solution purpose; 3. Design and development; 4. Demonstration and evaluation; 5. Communication (see Figure 1). DSRM focuses on developing a proof-of-concept level, as well as the resulting information technology discoveries, through the use of a select group of users to solve humanitarian and legacy problems. We began the DSRM sequence from a problem-centered initiation phase because, as previously stated, plagiarism of Balinese literary works occurs frequently as Balinese literary works develop. Based on these issues, we intend to create a Balinese document plagiarism detection system for Balinese literary works.

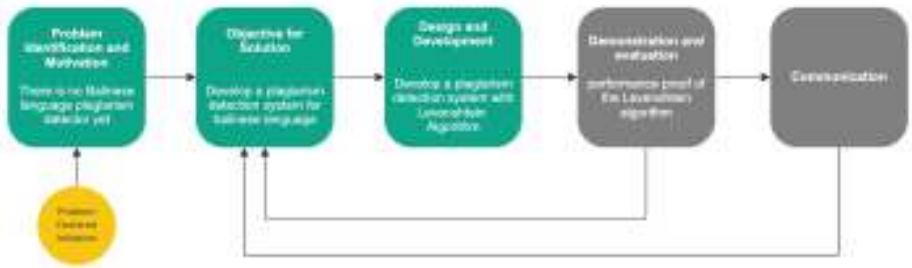

Figure 1 Research Phase

The data collection method used in this research is the library method The data used Figure 2 are basic Balinese words obtained from the article https://dictionary.basabali.org/Dictionary as many as 6271 basic words.

Figure 2 Basic Balinese Words

Source: https://dictionary.basabali.org/Dictionary

The basic words obtained will be processed using text mining before being used to detect document similarity. If the word has prefixes and suffixes or affixes like the Table 1 and Table 2 below, it will be deleted.

Table 1 Prefix List

No.	Prefix	Allomorph
1	n	ng , ny, n, m, nga
2	ma	-
3	pa	-
4	ka	-
5	sa	-
6	a	-
7	pra	-
8	pari	-
9	starch	-
10	maka	-
11	saka	-
12	kuma	-

Table 2 Suffix List

No	Suffix	Allomorphic
1	a	na
2	ang	nang, which
3	an	nan
4	in	nin
5	e	ne
6	ne	nne
7	n	-
8	ing	ning

The next method used is the Levenshtein Distance method to calculate document similarity, the Bastal algorithm used in stemming and Waterfall method for system development method (Febiawan et al., 2019; Hutagalung, 2019; Prasetya Wibawa et al., 2021; Rustamovna, 2021; Santoso et al., 2019; Yulianto et al., 2018).

3. RELATED RESEARCH

Table 3. Related Research

Indicators	Research I	Research II.
Author	(Sari et al., 2021)	(Febiawan et al., 2019)
Title of Research on	Plagiarism Detection Using the Levenshtein Distance Algorithm	Early Detection System of Plagiarism Using the Levenshtein Distance Algorithm
Research Object	at PSMTS ULM	Student Thesis Document
Research Methodology	Levenshtein Distance Method.	Levenshtein Distance Method.
Black	Black Box Testing	Box Testing

Research related to this research is (Sari et al., 2021), with the title "Plagiarism Detection Using the Levenshtein Distance Algorithm". This study discusses how to build an application to detect plagiarism in student thesis at PSMTS ULM.

Another study was taken from (Febiawan et al., 2019) with the title "Early Detection System of Plagiarism Using the Levenshtein Distance Algorithm". This study discusses early detection of thesis submissions for students majoring in informatics engineering, University of Muhammadiyah Magelang.

4. RESULTS AND DISCUSSION

Before detecting document similarity, the data obtained previously, namely, basic Balinese words will be processed in text mining first which has several stages including case folding, tokenizing, filtering and stemming (Abimanyu et al., 2020; Anggita S & Sanjaya ER, 2021; Sanjaya ER, 2021) . For example, the word Dogolan will be used as shown on Figure 3.

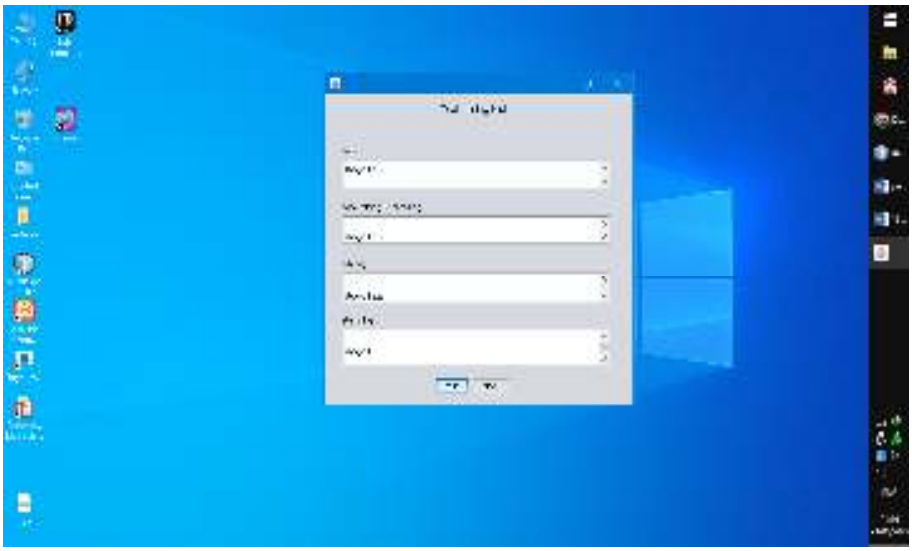

Figure 3 Text Mining Process

After passing through text mining, it will continue with document similarity detection using the Levenshtein Distance method to produce the percentage of similarity as shown in Figure 4.

Figure 4 Balinese Language Detection System

In the system testing stage, testing of the system built using black box testing and documents containing Balinese language will be carried out. System testing aims to analyze the system's capabilities based on the accuracy of the similarity obtained, processing time and CPU performance. The test will be carried out using several methods (Jampel et al., 2018; Sutramiani et al., 2021).

1) The first test will be carried out to test the level of accuracy using different documents and have the same number of words to test the CPU performance obtained to detect documents with a large number of words.

Table 4 First Testing

Original Document	Number of Words	Comparison Document	Number of Words	CPU Performance	Percentage	Processing Time
Test 1.txt	209	Test 1 (comparison).txt	209	3.1	24.8543	0:0:0:55
Test 2.txt	312	Test 2 (comparison).txt	312	3.1	21,1175	0:0:0:32
Test 3.txt	610	Test 3 (comparison).txt	610	5.6	27.8789	0:0:0:66
Test 4.txt	1023	Test 4 (comparison).txt	1023	4.8	24.2354	0:0:0:120
Test 5.txt	5007	Test 5 (comparison).txt	5007	33.0	24.8576	0:0:2:180
Test 6. docx	10108	Test 6 (comparison).docx	10108	33.0	24.8258	0:0:6:400
Test 7.docx	15143	Test 7 (comparison).docx	15143	33.0	34.5340	0:0:14:527
Test 8.docx	20262	Test 8 (comparison).docx	20262	33.0	23.9995	0:0:26:307
Test 9.docx	25020	Test 9 (comparison).docx	25020	33.0	24.2692	0:0:40: 91
Test 10.docx	30175	Test 10 (comparison).docx	30175	33.0	23.9565	0:0:60:616
Test 11.pdf	35184	Test 11 (comparison).pdf	35184	33.9	23.4353	0:1:18:432
Test 12.pdf	40244	Test 12 (comparison).pdf	40244	33.5	23.8220	0:1:9:495
Test 13.pdf	45052	Test 13 (comparison).pdf	45052	34.8	27.2364	0: 2:5:132
Test 14.pdf	50126	14 (comparison).pdf	50126	34.2	23.4048	0:6:13:524

The results of the comparison of the above documents, which have passed five times of testing, show that documents with a low percentage probability of the same

number of words produces similarity values, which are 23% to 34% and CPU performance on average 33% to 35%.

2) The second test will be conducted to test the stages and accuracy of the Levenshtein Distance method.

Table 8 Levenshtein Distance Method

			m	e	d	i	h
		0	1	2	3	4	5
m	1	0	1	2	3	4	
a	2	1	1	2	3	4	
d	3	2	2	1	2	3	
e	4	3	3	2	2	**3**	

In testing the stages of the Levenshtein Distance method the source string is used, namely made with the target string, i.e. medih produces a Levenshtein Distance, which is 3. The string made with medih requires 3 operations, namely two substitutions on "a" to "e" and "e" to "i" and one insertion "h" on string made to string medih. Based on the Levenshtein Distance resulting from the above calculation, the following equation is used to determine the accuracy of the percentage of similarity using the Levenshtein Distance method from string made to string medih.

$$
\begin{aligned}
& diff = 3 \\
& Max(CS, ST) = 5 \\
&= \left(1 - {}^{3}/_{5}\right) * 100 \\
&= \left({}^{5}/_{5} - {}^{3}/_{5}\right) * 100 \\
&= 40\%
\end{aligned}
$$

The results of the test using the system can be seen in Figure 5, which shows that the words made and medih get a similarity value of 40% the same as the manual calculations carried out.

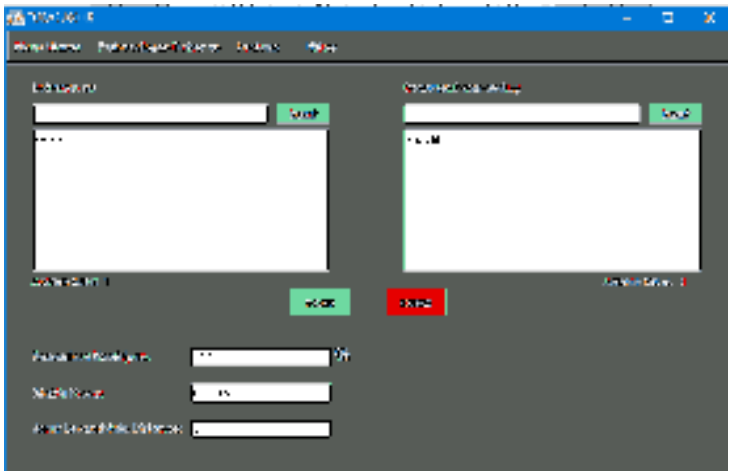

Figure 5 Testing Result with System

5. CONCLUSION

Based on the description and results of the analysis that has been carried out during the development of the Balinese document similarity detection system using the Levenshtein Distance method, it can be concluded that:

1. The Balinese document similarity detection system using the Levenshtein Distance method was successfully built using the Levenshtein method distance.
2. In testing using data in the form of a collection of Balinese language units, the Levenshtein distance algorithm produces a high similarity value for documents with similar content, namely, 100% while for documents with different contents, the percentage is 23% to 34%.
3. In tests that use two documents with the same content and the same number of words, the CPU performance is 36.1% and an average of 33%, while tests that use two documents that are similar to the same content get CPU performance, namely, 33.3% and the average is, 30% and the test uses the original document which has more words than the comparison document with the test using the original document which has less word count than the comparison document gets the same CPU performance, namely, 33% to 36%.

REFERENCES

Abimanyu, C. G., ER, N., & Karyawati, A. A. I. N. E. (2020). BALINESE AUTOMATIC TEXT SUMMARIZATION USING GENETIC ALGORITHM. *JITK (Jurnal Ilmu Pengetahuan Dan Teknologi Komputer)*, *6*(1), 13–20. https://doi.org/10.33480/JITK.V6I1.1344

Anggita S, N. P. A. S., & Sanjaya ER, N. A. (2021). Location Named-Entity Recognition using Rule-Based Approach for Balinese Texts. *JELIKU (Jurnal Elektronik Ilmu Komputer Udayana)*, *9*(3), 435. https://doi.org/10.24843/JLK.2021.V09.I03.P15

Febiawan, M. H., Setiawan, A., & Primadewi, A. (2019). Sistem Pendeteksi Dini Plagiarisme Menggunakan Algoritma Levenshtein Distance. *Jurnal Komtika (Komputasi Dan Informatika)*, *3*(1), 18–27. https://doi.org/10.31603/KOMTIKA.V3I1.3464

Habibi, I., & Munir, R. (2009). The Balinese Unicode Text Processing. *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, *1*(1). https://doi.org/10.22146/ijccs.19

Hutagalung, C. (2019). Pendeteksian Plagiarisme Skripsi Menggunakan Algoritma Levenshtein Distance Berbasis Web |. *Journal of Computer System and Informatics (JoSYC)*, *1*(1). https://ejurnal.seminar-id.com/index.php/josyc/article/view/34

Jampel, I. N., Indrawan, G., & Widiana, I. W. (2018). Accuracy analysis of Latin-to-Balinese script transliteration method. *International Journal of Electrical and Computer Engineering*. https://doi.org/10.11591/ijece.v8i3.pp1788-1797

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems*, *24*(3), 45–77. https://doi.org/10.2753/MIS0742-1222240302

Pramartha, C. R. A., & Gede Dwidasmara, I. B. (2014). The composition approach non-QWERTY keyboard for Balinese script. *2014 IEEE Canada International Humanitarian Technology Conference - (IHTC)*, 1–4. https://doi.org/10.1109/IHTC.2014.7147554

Prasetya Wibawa, A., Nu, M., Hakim, man, Semarang No, J., Lowokwaru, K., Malang, K., & Timur, J. (2021). STEMMING BAHASA JAWA MENGGUNAKAN DAMERAU LEVENSHTEIN DISTANCE (DLD). *JURNAL TEKNIK INFORMATIKA*, *14*(1), 22–27. https://doi.org/10.15408/JTI.V14I1.15010

Rustamovna, A. U. (2021). Understanding the Levenshtein Distance Equation for

Beginners. *The American Journal of Engineering and Technology*, 134–139. https://medium.com/@ethannam/understanding-the-levenshtein-distance-equation-for-beginners-c4285a5604f0

Sanjaya ER, N. A. (2021). Implementasi Latent Dirichlet Allocation (LDA) untuk Klasterisasi Cerita Berbahasa Bali. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, *8*(1), 127. https://doi.org/10.25126/JTIIK.0813556

Santoso, P., Yuliawati, P., Shalahuddin, R., & Wibawa, A. P. (2019). Damerau Levenshtein Distance for Indonesian Spelling Correction. *Jurnal Informatika*, *13*(2), 11–15. https://doi.org/10.26555/JIFO.V13I2.A15698

Sari, Y., Khatimi, H., & Awlia Fajrin, R. (2021). DETEKSI PLAGIARISME MENGGUNAKAN ALGORITMA LEVENSHTEIN DISTANCE. *Jurnal Teknologi Informasi Universitas Lambung Mangkurat (JTIULM)*, *6*(1), 31–38. https://doi.org/10.20527/JTIULM.V6I1.66

Sutramiani, N. P., Suciati, N., & Siahaan, D. (2021). MAT-AGCA: Multi Augmentation Technique on small dataset for Balinese character recognition using Convolutional Neural Network. *ICT Express*, *7*(4), 521–529. https://doi.org/10.1016/J.ICTE.2021.04.005

Yulianto, M. M., Arifudin, R., & Alamsyah, A. (2018). Autocomplete and Spell Checking Levenshtein Distance Algorithm To Getting Text Suggest Error Data Searching In Library. *Scientific Journal of Informatics*, *5*(1), 75. https://doi.org/10.15294/SJI.V5I1.14148