Development of natural language processing on morphology-based Minangkabau language stemming algorithm

Rini Sovia[1], Sarjon Defit[1], Yuhandri[1], Sulastri[2]
[1]Department of Information Technology, Faculty of Computer Science, Universitas Putra Indonesia YPTK, Padang, Indonesia
[2]Department of Cultural Sciences, Faculty of Languages and Literature, Andalas University, Padang, Indonesia

Article Info

Article history:

Received Oct 17, 2022
Revised Mar 13, 2023
Accepted Mar 24, 2023

Keywords:

Minangkabau language
Natural language processing
Minangkabau language
Stemming algorithm
Stemming algorithm translation

ABSTRACT

Minangkabau language (ML) is one of the daily communication tools used by the people of West Sumatra, Indonesia. ML is a challenge in communicating. The ML language translation process is necessary to facilitate communication. This study aims to build a translation system for ML into Indonesian by developing the concept of natural language processing (NLP). NLP development adopts the performance of morphology-based Minangkabau language stemming algorithm (MLSA) which can separate basic words with affixes and endings. The research dataset adopts 600 basic ML words sourced from the big Minangkabau dictionary. The results of this study provide analytic output that can translate ML into Indonesian well. These results are presented based on the testing process on basic word input with an accuracy rate of 97.16% and based on text documents of 91.65%. Thus, the MLSA performance process presents the accuracy of the translation process. Based on these results, this research contributes to developing a stemming algorithm model in carrying out the process of removing prefixes, inserts, and suffixes in the Minangkabau language. Overall, this research can be useful as a tool for translating the ML into Indonesian.

Corresponding Author:

Rini Sovia
Department of Information Technology, Faculty of Computer Science
Universitas Putra Indonesia YPTK Padang
Padang, Indonesia
Email: rini_sovia@upiyptk.ac.id

1. INTRODUCTION

A language is a tool used in sociolinguistic communication to interpret the form of a person’s behavior [1]. Language can function as a medium of communication between humans [2], [3]. Language is also a medium of communication between humans in an integrated manner [4]. The development of language today has produced various kinds of language forms that are spread in several regions [5]. One of the languages in question is the Minangkabau language (ML) [6]. ML is a daily communication tool by the people of West Sumatra, Indonesia [7]. ML is not only used as a popular language in West Sumatera but has also developed due to the urbanization of the Minangkabau community in various regions globally [8]. Based on the facts, it shows that the urbanization that has occurred has affected the maintenance of the ML in Jakarta [9]. Not only that, but ML is also a problem for tourists visiting West Sumatra in communicating with local communities [10].

These problems become the focus of this research to present a system that can perform the conversion process on ML. This is based on the motivation to know and understand ML [11]. The resulting system will have the aim of introducing ML effectively [12]. This process will also be used in reviewing

Minangkabau traditional culture through language [13]. To present this, the concept of natural language processing (NLP) is used to develop a system capable of processing, processing, analyzing, and managing human language [14]. The concept of NLP can obtain structural, syntactic, and semantic rules in a computer environment so that it can perform analysis [15]. The term NLP refers to a symbolic communication system (oral, signature, or written) that develops with planning and design on an analytical model [16]. The development of NLP has presented a series of learning processes by showing the results of progress in dealing with a problem [17]. The concept of NLP provides semantic analysis which is an important feature in the context of sentences or paragraphs [18]. NLP can contribute greatly to the problem of language grouping by providing dynamic updates [19]. In general, NLP can produce analytical models that are applied to a program in completing tasks [20].

Previous studies have explained that NLP can extract information about English translation features [21]. The development of NLP in other cases has also shown results that can extract information from free-text narratives written by various health care providers [22]. The NLP can play a good role in modeling the Chinese language with fairly accurate results [23]. NLP can identify with qualitative analysis of text data [24]. The application of text mining techniques and NLP can perform analysis of text data by producing optimal output [25]. NLP can convert different word forms into standard root forms [26]. The application of NLP in ML has been presented in previous studies by performing morphological and syntactic analyzes [27]. In the same study, NLP was able to present the correct language control system and grammar in the ML [28]. The application of the NLP concept by adopting the performance of an artificial neural network (ANN) can carry out a translation process that focuses on the ML and Indonesian language using 13,761-word pairs resulting in an average precision output of 83.55% [29].

The development of NLP can be presented on the performance of the stemming algorithm in conducting analysis. A stemming algorithm is a computational process of removing all suffixes and prefixes from a word to produce stems or roots [30]. Stemming algorithms are developed for various applications that implement artificial intelligence based on natural language morphology rules [31]. The stemming algorithm that has been developed contributes quite well to separating basic words from prefixes, infixes, suffixes, and combinations (confixes) [32]. This separation process includes the removal of affixes [33]. The stemming algorithm is used to process the analysis of differences in human language [34]. The stemming process performs the process of reducing the inflection or the resulting term into the main word, root, or original form [35]. The stemming process adds reduplication rules, prefixes, and suffixes to increase the accuracy of each word [36]. The stemming process can perform analysis in information retrieval, speech tagging, syntactic parsing, and machine translation [37]. Forms of words that can be used to detect text in searching, searching, mining, summarizing, classifying, and making decisions [38]. The stemming algorithm is used at the text pre-processing stage to improve the performance of text applications [39]. The results of this process can be used for plagiarism, spell-checking, and search engines [40].

Based on the previous explanation, this study presents the process of NLP analysis by adopting the performance of text morphological processes in the process of translating ML into Indonesian. The NLP analysis process was developed using the morphology-based Minangkabau language stemming algorithm (MLSA) model. This algorithm can provide updates in the performance of the stemming process to model basic words generated from ML. The performance of the MLSA also maximizes the stemming process to automatically delete words such as prefixes, inserts, and suffixes. This update can make a more optimal level of NLP performance in automating ML translations into Indonesian. The contributions made in this study provide development of a stemming algorithm model that can be applied to the ML. With this contribution, this research has a positive impact on language translation.

2. METHOD

This study presents the process of translating the ML into Indonesian by developing the concept of NLP. The role of the performance of the stemming algorithm will be maximized to present the basic word. The development process can be presented in the research framework depicted in Figure 1. Figure 1 is the process of developing NLP in the Minangkabau language translation into Indonesian. Stages of the process begin with the process of morphological analysis using the performance of the UG18 algorithm. The results obtained will be carried out with a steaming process using morphological techniques to find each basic word. The basic word results generated will be stored in the database. The stored data become knowledge-based which will later be used in the translation process.

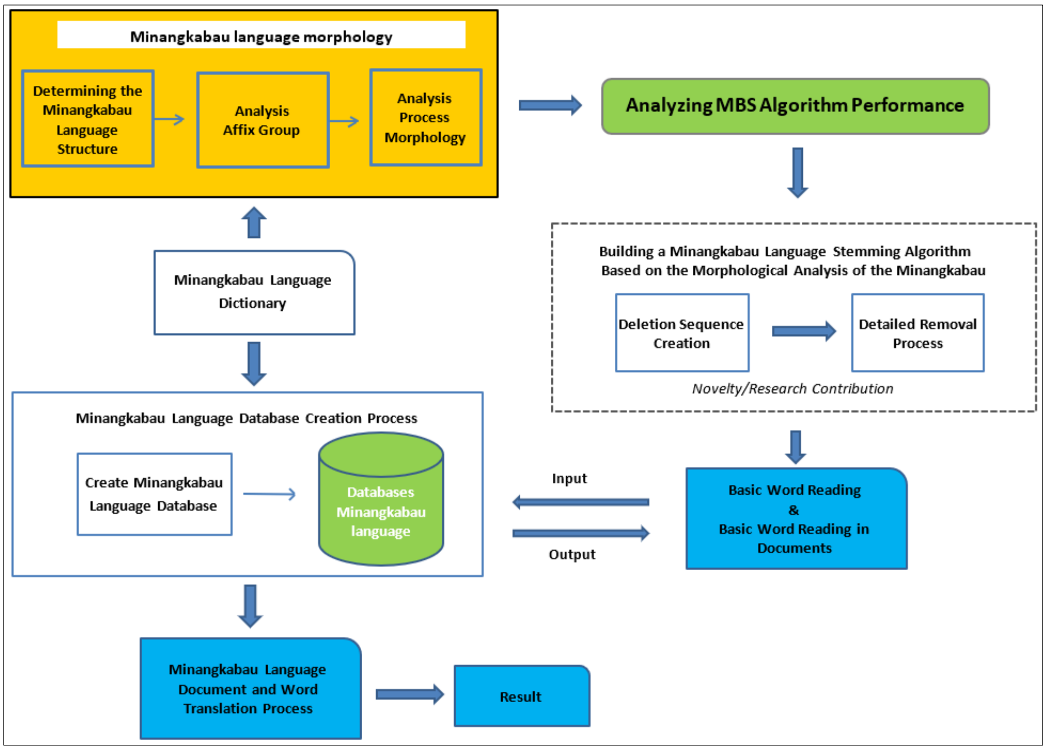

Minangkabau language morphology
Determining the Minangkabau Language Structure
Analysis Affix Group
Analysis Process Morphology
Analyzing MBS Algorithm Performance
Minangkabau Language Dictionary
Building a Minangkabau Language Stemming Algorithm Based on the Morphological Analysis of the Minangkabau
Deletion Sequence Creation
Detailed Removal Process
Novelty/Research Contribution
Minangkabau Language Database Creation Process
Create Minangkabau Language Database
Databases Minangkabau language
Input
Output
Basic Word Reading & Basic Word Reading in Documents
Minangkabau Language Document and Word Translation Process
Result

Figure 1. Research framework

2.1. Related work

Several previous studies that correlate with the development of NLP can be presented for a review of the concept and performance of the method in presenting outputs. The review was carried out to ensure the novelty produced by this research will have an impact on the translation process. The related research can be presented in Table 1.

Table 1. Overview of NLP concepts and method performance

No	Author and Year	Method	Result
1.	Nzeyimana, 2021 [41]	Dataset development, morphological analysis, and classification	Morphological disambiguation of verb forms using the model's maximum entropy on a new crowd-source stemming data set.
2.	Yucebas and Tintin, 2021 [15]	Finite-state machine (FSM) design and matching and control	GovdeTurk is a tool for stemming, morphological labeling and verb negation for Turkish language, finding word roots by removing inflectional suffixes in longest matching strategy.
3.	Sibi *et al.*, 2020 [42]	Nazief and Adriani algoritma algorithm	Knowing the syllables of a word that gets prefixes and word suffixes in the Indonesian punctuation system (SIBI) with Indonesian morphology.
4.	Mulyana *et al.*, 2019 [43]	Energy conscious scheduling (ECS) algorithm and NECS algorithm	Grouping of affixes into 6 groups deletion without using beheading rules using a data dictionary.
5.	Yusliani *et al.*, 2019 [44]	Single process engineering (SP), Multiprocess (MP) and ECS algorithm	Optimizing the speed of wanting to get stemming using multi-processing (MP) using ECS.
6.	Alotaibi and Gupta, 2018 [45]	Measuring lexical and structural distances, formation of morphology class, and identify query relative variants.	Cognitive language independent stemming technique that groups words morphologically in a corpus without linguistic or manual knowledge.

Table 1 presents previous research in the analysis process using the stemming algorithm. The results presented above can group words morphologically with a good output. Based on this, this study will also develop a stemming algorithm to carry out the process of translating ML into Indonesian.

The development of the stemming algorithm in NLP for the ML translation process will be developed using a morphological process. The stemming algorithm by adopting the performance of the morphological morpheme technique is used to maximize the ML translation process. The translation process is presented in several stages. First, the ML stemming process is used to find the basic words contained in the Indonesian dictionary. If the found word becomes the root word, the steaming process stops. However, if the word is not found, then the process of deleting the prefix is carried out and continues at the next stage. Second, the validation process is used to ensure that the basic words are contained in the dictionary. The validation process will also remove words that contain suffixes. In this case, if the validation process does not find the word in question, then the process is continued at the next stage. Third, is the process of measuring each word that has an insert. This process is carried out to remove words that have inserts in each validated word. If the measurement process is found, then output the result and if not, then the process stops. The performance of the morphology based MLSA can be seen in Figure 2.

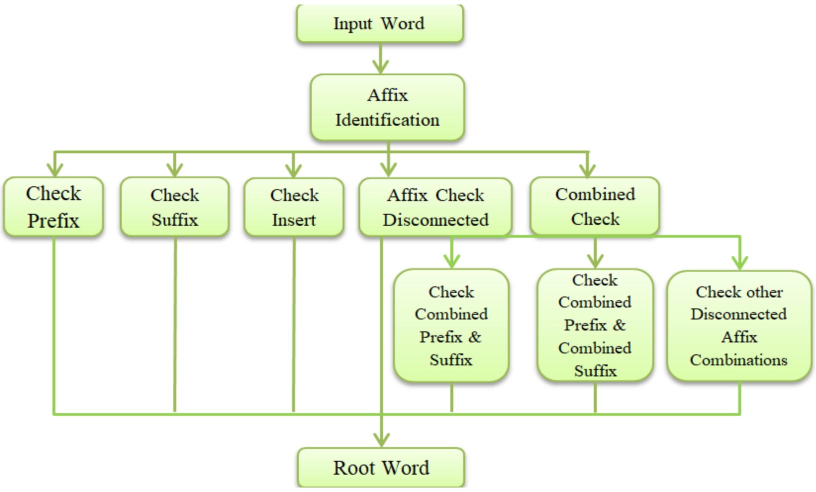


Figure 2. Morphology-based Minangkabau language stemming algorithm

Figure 2 illustrates the performance of the MLSA in the process of translating the ML into Indonesian. MLSA performance can perform the process of eliminating every basic word. The elimination process is described in the form of prefix deletion, suffix, suffix, insertion, and combined deletion. The results from MLSA provide output in the form of basic words that can be used as a process of translating the ML into Indonesian.

2.2. Material

2.2.1. Natural language processing

NLP is a branch of artificial intelligence that is limited to natural (linguistic) language. Natural language can be interpreted as a discipline of several other branches of science such as computer science, artificial intelligence, and cognitive psychology. The NLP on the other hand, refers to software tools equipped with features related to processing languages, [45]. The discussion includes speech (speech segmentation), text segmentation (text segmentation), word class marking (part-of-speech tagging), and meaning control (word sense disambiguation) [46].

2.2.2. Based stemming algorithm

Based stemming algorithm (MBS) is a combined algorithm for grouping affixes [47]. The grouping of affixes in the MBS algorithm consists of prefixes without morphophonemic processes, prefixes with morphophonemic processes, combined prefixes, suffixes, possessive pronouns, and particles [48]. MBS is a stemming algorithm that was developed without using the beheading rule but has a high level of truth [49]. MBS can present results with a fairly low error rate in describing over stemming and under stemming [50].

3. RESULTS AND DISCUSSION

The discussion in this study will carry out several stages of analysis to present the desired output. These stages can be described in the research framework that has been described previously. The performance of the developed stemming algorithm is expected to provide an effective and efficient analysis process in translating the ML into Indonesian.

3.1. Preprocessing stage

The preprocessing stage is the initial stage in the analysis of the NLP concept. The purpose of preprocessing is to convert raw data into a more accessible form. Generally, a document has a capital letter presentation form at the beginning of a paragraph or the beginning of a sentence, containing punctuation marks, such as periods, and commas. This condition causes the data cannot be managed directly, which is caused by unstructured data. This condition needs to be done at the preprocessing stage which consists of case folding, tokenizing, stopword removal, and stemming as shown in Figure 3.

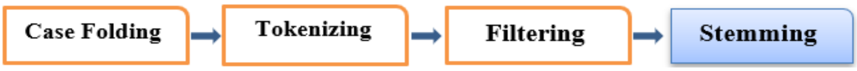


Figure 3. Pre-processing stage

Figure 3 is the pre-processing stage presented in the early stages of the analysis of the translation process. These stages include the process of case folding, tokenizing, filtering, and stemming. The process carried out presents the development of the MBS algorithm in performing ML translation operations into Indonesian. The development of the stemming algorithm can be presented in the next section:

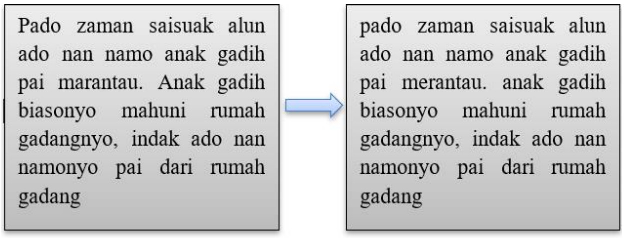
3.1.1. Case folding

Case folding is a process that is often neglected in text preprocessing [51]. Basically case folding is the simplest and most effective process in text mining [52]. The function of this folding case is to convert all uppercase letters into lowercase letters [53]. The development of case folding can be presented in the Pseudocode 1 algorithm. Pseudocode 1 is an algorithm developed to perform the case folding process on each word that is input from the ML. This process will capitalize each word. The case folding process will produce a product with lowercase letters. The output of the case folding process can be presented in Figure 4.

Figure 4 shows the results of the case folding stages. The results of the process provide output in the form of changing all letters to lowercase, characters such as periods, and commas are still contained in the sentence. Based on the results presented, the development of case-folding can be applied to the ML.

Pseudocode 1. Case folding algortihm

```
Input   : Word, Doc
Output  : Word, Doc
        Function Text_Processing (text) case folding
        Foreach ($array delete as $ key ”$kata”, “$Dokumen”){
        $kataRegex=preg_replace {‘/ (‘$kata, $dokumen’);
        Echo stemming Minang(strtolower($kataRegex),
        $jmlstem, $under).’';
            }
}
```



Pado zaman saisuak alun ado nan namo anak gadih pai marantau. Anak gadih biasonyo mahuni rumah gadangnyo, indak ado nan namonyo pai dari rumah gadang

pado zaman saisuak alun ado nan namo anak gadih pai merantau. anak gadih biasonyo mahuni rumah gadangnyo, indak ado nan namonyo pai dari rumah gadang

Initial Sentence

Folding Case Results

Figure 4. Case folding process results

3.1.2. Tokenizing

Tokenizing is a process step used to separate text into words [54]. This process aims to divide sentences into words with the smallest units [55]. The tokenizing process can also be used to remove numbers, characters, and punctuation marks such as periods (.), commas (,), and exclamation points (!) [56]. The development of case-folding can be presented in Pseudocode 2. Pseudocode 2 presents the tokenizing process which is presented in an algorithm developed to continue the process of the case folding process. The tokenizing algorithm works to ensure the division of sentences into word units that can be read at a later stage. Based on the experiments that have been carried out, the tokenizing output results will be reused at the stage of the filtering process.

Pseudocode 2. Tokenizing algortihm

```
Input   : Word, Doc
Output  : Word, Doc
        Function Text_Processing (text) case folding.
        Foreach ($array delete as $ key” $kata”, “$Dokumen”){
        $kataRegex=preg_replace {‘/ (‘$kata, $dokumen’);
        Echo stemming Minang(strtolower($kataRegex),}
        $jmlstem, $under).’';
        $split= exploded (‘’, $kalimat);
        $kataArray= []
        Foreach ($split as $key=$ kata) {
        $kataArray [] =$kata;}
}
```

3.1.3. Filtering/stopword removal

Stopword removal/filtering is a text mining analysis process in the process of selecting words that are not important [56]. These unimportant words are words that have no meaning such as, which, and, in, from, or, to, and so on which will later be stored in a stoplist/stopword [57]. The development of the filtering process can be presented in the algorithm in Pseudocode 3. Pseudocode 3 presents the development of algorithms in the filtering process. The results given from this process can validate words that have no meaning and meaning in the ML. These results will later be used as input in conducting the analysis process by developing a stemming algorithm using a morphological process.

Pseudocode 3. Filtering/stopword removal algortihm

```
Input   : Word, Doc
Output  : Word, Doc
        Function Text_Processing (text) case folding
        Foreach ($array delete as $ key ”$kata”, “$Dokumen”){
        $kataRegex=preg_replace {‘/ (‘$kata, $dokumen’);
        Echo stemming Minang(strtolower($kataRegex),
        }
        $jmlstem, $under).’';
        $split= exploded (‘’, $kalimat);
        $kataArray= []
        Foreach ($split as $key=$kata) {
        $kataArray [] =$ kata
        }
        $stopword= array ()
        $getremoving= $conn query (‘Select *from stop removing’);
        Foreach ($getremoving as $ stop) {
        $stopword [] =$stop{‘kata’};
        }
             $ arrayHapus= array_diff ($kataArray, $stopword);
```

3.1.4. Minangkabau language stemming algorithm

The development stemming algorithm in the ML translation process was developed using a morphological process. This process presents the beheading of affixes in the ML. The affixes consist of groups of prefix affixes (ba-1, ba-2, maN, pa, ta-, no, di, sa, ka, basi, standard), insertion affixes (-il,-al,-ar, -am, iŋ), suffix groups (an,-kan, and lah), compound groups (pan.-an, ba-..-an, paN-..-y, and ka..-an). The presentation of these types of ML affixes can be seen more clearly in Table 2. Table 2 presents the types of affixes from the ML for stemming processes using morphological operations. This steaming process will be able to cut off any words that have affixes which are presented in Table 2. The development of stemming algorithms with morphological processes can be presented in Pseudocode 4.

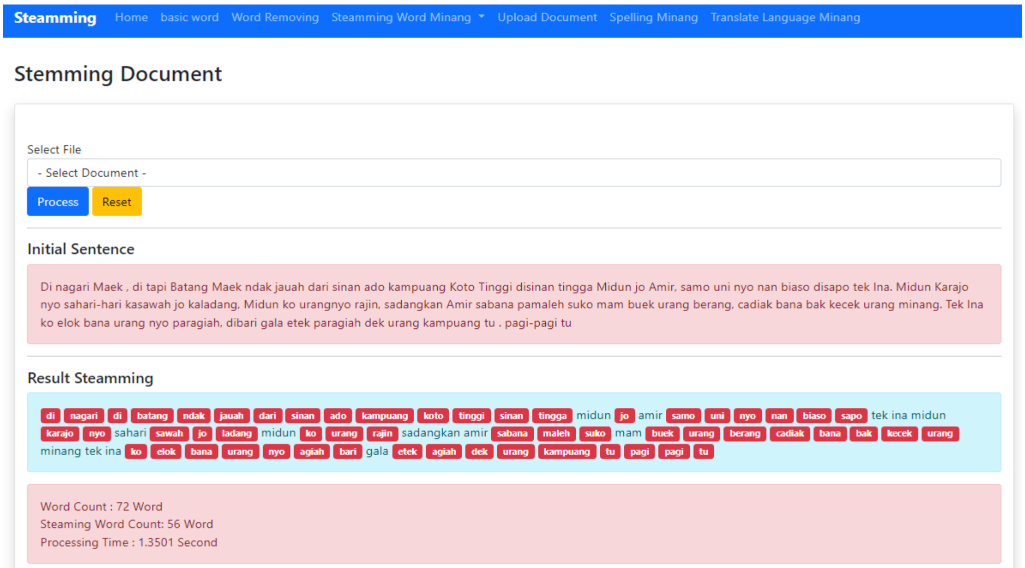
The development of the stemming algorithm by adopting a morphological process presents the rules used in chopping off the affixes obtained in the previous process. The performance of this algorithm has been presented in Figure 2 to present the basic words of the ML itself. If in the steaming process the basic word has been obtained, the word will be stored in a database which will later be forwarded in the process of translation into Indonesian. This translation process can be implemented on a system built which is presented in Figure 5.

Table 2. Types of Minangkabau language affixes

No.	Word group	Word type
1.	Prefix	ba-1, ba-2, maN, paN-, pa-,ta, no, sa, baku, baka, basi, ka, bapa, tapa, maN pa, sa pa
2.	Infix	-il, -al, -ar, -am, iŋ
3.	Suffix	-an, -kan,I, dan lah
4.	Affixes cut off	ka..an, ka..no, paN..an
5.	Combination of prefix and suffix	Combination of prefix and suffix (ba Kan, ba-i, no-Kan, pa-Kan, ba-lah, baku-lah, basi-lah), combined prefix and suffix (MaN- pa- Kan, no- pa- Kan, No-sa-Kan, sa-paN, di-pa-sa-Kan)
6.	Combined prefix and combined suffix	maN..pa.., Kanlah, maN..sa.., Kanlah dipa.., Kanlah, disa.., Kanlah, baku.., lah, basi..lah, sapaN..,lah
7.	Combined affixes interfere with other affixes	ba2..ka..an, ba2..paN..an, sa..paN.an

Pseudocode 4. Minangkabau language stemming algorithm

```
Input   : Word, Doc
Output  ; Word, Doc
  – Search for the word to be deleted in the dictionary, if found then the word is the root
    word, and the process stops;
  – Look for the prefix, if there is perform the initial deletion, then check in the
    dictionary and display it, if it is not there;
  – Look for the suffix and, if it exists, delete the suffix, then check in the dictionary
    and display it, if it doesn’t exist;
  – Look for the suffix kan, if there is a deletion of the suffix kan", then check in the
    dictionary, and display it, if not there;
  – Look up my suffix and lah, if there is a suffix deletion, then check in the dictionary
    and lah, and display it, if it doesn’t exist;
  – Look for one of the inserts such as il, if there is a deletion of the il insert, then
    check in the dictionary, and display it, if it does not exist;
  – Look for the insert al, if there is delete the insert al, then check the dictionary,
    display it, if not there;
  – Search for insertion or, if there is a deletion of insertion al, then check the
    dictionary, display it, if there is none;
  – Looking for am insertion, if there is no deletion of am insertion, then check the
    dictionary, display it, if not there;
  – Look for the insert iŋ, if there is a deletion of the insert iŋ, then check the
    dictionary, and display it, if it doesn’t exist;
  – Look for broken affixes, if there is a deletion of insertions, then check the
    dictionary, and display it, if there is none;
  – Searching for confixes, if there are any, delete confixes, then check in the dictionary,
    and display them, if they don’t exist;
If all the steps have been completed it does not also work, then the initial word but as
the root word. Process completed.
```



Stemming Document

Select File

- Select Document -

Initial Sentence

Di nagari Maek , di tapi Batang Maek ndak jauah dari sinan ado kampuang Koto Tinggi disinan tingga Midun jo Amir, samo uni nyo nan biaso disapo tek Ina. Midun Karajo nyo sahari-hari kasawah jo kaladang, Midun ko urangnyo rajin, sadangkan Amir sabana pamaleh suko mam buek urang berang, cadiak bana bak kecek urang minang. Tek Ina ko elok bana urang nyo paragiah, dibari gala etek paragiah dek urang kampuang tu . pagi-pagi tu

Result Steamming

di nagari di batang ndak jauah dari sinan ado kampuang koto tinggi sinan tingga midun jo amir samo uni nyo nan biaso sapo tek ina midun karajo nyo sahari sawah jo ladang midun ko urang rajin sadangkan amir sabana maleh suko mam buek urang berang cadiak bana bak kecek urang minang tek ina ko elok bana urang nyo agiah bari gala etek agiah dek urang kampuang tu pagi pagi tu

Word Count : 72 Word
Steaming Word Count: 56 Word
Processing Time : 1.3501 Second

Figure 5. Implementation of the ML translation system

Figure 3 is an implementation of the development of a stemming algorithm with a morphological process on the system that has been designed. The results of the system work have been able to carry out stemming operations well on translations based on inputted words and documents. The output of the system can present information on the number of words that have affixes with a fairly good performance. To ensure that the stemming algorithm that has been developed provides optimal results, a testing process is needed to validate the performance of the algorithm. The validation results obtained are presented in Tables 3 and 4.

Table 3. Test result on the world

No	Group	Word total	Word stem	Accuracy
1.	Prefix	387	381	98.00
2.	Insert	11	10	91.00
3.	Suffix	96	90	94.00
4.	Disconnected affix	59	57	97.00
5.	Combination of prefix and suffix	18	17	94.00
6.	Combination of prefix and combination of suffix	24	23	96.00
7.	Other disconnected affixes	5	5	100.00
	Average			97.16

Table 4. Test results on document

No	Document	Word total	Word stem	Accuracy
1.	Documen 1	199	180	90.00
2.	Documen 2	72	65	90.00
3.	Documen 3	112	100	89.00
4.	Documen 4	1,403	1,320	94.00
5.	Documen 5	394	372	94.00
6.	Documen 6	453	435	96.00
7.	Documen 7	1,722	1,604	93.00
8.	Documen 8	518	480	93.00
9.	Documen 9	193	172	89.00
10.	Documen 10	477	464	97.00
11.	Documen 11	399	310	78.00
12.	Documen 12	507	409	81.00
	Average			91.65

Table 4 presents the validation process based on the work system with the word input process manually. Validation results are based on testing several forms of sentences used. The output of the translation process from the development of this stemming algorithm presents a validation rate of 97.16%. Tests based on documents were also carried out on some of the samples used and obtained a validation result of 91.65%. To measure the accuracy of the results of the performance of the stemming algorithm developed in the translation process, it can be presented in (1) [58].

$$Accuracy = \frac{ASK + ASD}{2} = \frac{97.16\% + 91.65\%}{2} = 94.40\% \quad (1)$$

Based on the process of measuring the level of accuracy using (1), the results obtained are 94.40%. These results have proven that the process of translating the ML into Indonesian has given optimal results. The development of a stemming algorithm using a morphological process is in line with previous research in developing a standard-based stemming algorithm [59]. Other developments have also provided better results based on the stemming enhance confix stripping (ecs) and new enhance confix stripping (necs) algorithms [60], [61].

Based on the performance of the results of the stemming algorithm developed using a morphological process, it has been able to provide optimal results in translating ML into Indonesian. These results are based on the morphology based MLSA model which was developed in the steaming process to remove prefixes, inserts, and suffixes. The MLSA has also been able to provide updates to the performance of the stemming process in modeling basic words generated from ML. Thus, the MLSA makes a novelty in the concept of NLP which uses natural language processing. Based on these findings, the contribution made in this study presents a stemming algorithm model developed to be able to translate the Minangkabau language. With these contributions, this research has an effective impact on developing the concept of NLP in language translation.

4. CONCLUSION

The development of this stemming algorithm has worked optimally in the process of translating the ML into Indonesian. This algorithm can work by cutting input words based on insertions and affixes. The performance results provide accuracy in the translation process with an accuracy of 94.40%. Based on these results, this algorithm can be effectively used to translate the ML into Indonesian. Not only that, but this algorithm is also able to present new knowledge about stemming algorithms.

ACKNOWLEDGEMENTS

Thanks to Mrs. Dr. Sulastri, M.Hum. as a Lecturer at the Faculty of Cultural Sciences, Andalas University who was used as one of the experts in validating the results of this study based on the appointment letter of the Faculty of Computer Science, No. 003/ PSDTI/UPI-YPTK/IP/VIII/2021, regarding the appointment of experts in this study.

REFERENCES

[1] T. Thamrin, “The language attitudes of Minangkabau people towards Minangkabau and Indonesian language,” *International Journal of Language Teaching and Education*, vol. 2, no. 2, pp. 157–175, Aug. 2018, doi: 10.22437/ijolte.v2i2.5065.

[2] R. Fowler, *Understanding language: an introduction to linguistics*. Routledge, 2022.

[3] F. D. Varennes, *Language, Minorities and Human Rights*. Brill, 2021.

[4] M. Fröhlich, C. Sievers, S. W. Townsend, T. Gruber, and C. P. V. Schaik, “Multimodal communication and language origins: integrating gestures and vocalizations,” *Biological Reviews*, vol. 94, no. 5, pp. 1809–1829, Oct. 2019, doi: 10.1111/brv.12535.

[5] A. Syakur, R. Purba, D. Chaniago, H. Herman, S. O. Manullang, and M. Muhammadiah, “Variety of languages on the status of Facebook users,” *Proceedings of the 3rd International Conference of Science Education in Industrial Revolution 4.0, ICONSEIR 2021*, 2021, doi: 10.4108/eai.21-12-2021.2317485.

[6] R. K. Tarihoran and D. Widayati, “Lexicostatistics of Toba language, Sibolga language, and Minangkabau language,” *Budapest International Research and Critics Institute (BIRCI-Journal): Humanities and Social Sciences*, vol. 5, no. 3, pp. 18318–18328, 2022.

[7] T. Oswari, E. Hastuti, and R. Chandra, “Minangkabau language learning based on android application,” in *4th International Conference on Arts Language and Culture (ICALC 2019)*, 2020, pp. 277–284, doi: 10.2991/assehr.k.200323.033.

[8] W.S. Hasanuddin, “Minangkabau language greeting system in creative text works: A case study on modern Indonesian fiction Minangkabau local color and lyrics of popular modern Minangkabau songs,” *Humanus*, vol. 19, no. 2, p. 161, Oct. 2020, doi: 10.24036/humanus.v19i2.108619.

[9] E. Hastuti and T. Oswari, “The influence of effort on the Minangkabau language maintenance in Jakarta,” *Proceedings of the 4th International Conference on Arts Language and Culture (ICALC 2019)*, 2020, doi: 10.2991/assehr.k.200323.029.

[10] R. Susanti and F. Syahar, “Tour De Singkarak, West Sumatra event sustainable marketing and tourism,” *International Journal of Tourism, Heritage and Recreation Sport*, vol. 1, no. 1, pp. 32–38, Jun. 2019, doi: 10.24036/ijthrs.v1i1.15.

[11] T. Thamrin, “Minangkabau language: use and attitudes.” La Trobe, 2023.

[12] A. Rengganis, S. Sarkum, I. R. Munthe, and I. Purnama, “Android based Minang language using UCD method (user centered design): tour guide application,” *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, vol. 7, no. 2, pp. 149–154, Sep. 2018, doi: 10.32736/sisfokom.v7i2.554.

[13] S. F. Dewi *et al.*, “The role of culture in cross-cultural marriage among Minangkabau women,” *Journal of International Women’s Studies*, vol. 20, no. 9, pp. 68–82, 2019.

[14] Y. Kang, “Natural language processing (NLP) in management research: a literature review,” *Journal of Management Analytics*, vol. 7, no. 2. pp. 139–172, 2020, doi: 10.1080/23270012.2020.1756939.

[15] S. Yucebas and R. Tintin, “Govdeturk: a novel turkish natural language processing tool for stemming, morphological labelling and verb negation,” *International Arab Journal of Information Technology*, vol. 18, no. 2, pp. 148–157, 2021, doi: 10.34028/IAJIT/18/2/3.

[16] K. Darwish *et al.*, “A panoramic survey of natural language processing in the Arab world,” *Communications of the ACM*, vol. 64, no. 4, pp. 72–81, Apr. 2021, doi: 10.1145/3447735.

[17] D. Ofer, N. Brandes, and M. Linial, “The language of proteins: NLP, machine learning & protein sequences,” *Computational and Structural Biotechnology Journal*, vol. 19, pp. 1750–1758, 2021, doi: 10.1016/j.csbj.2021.03.022.

[18] D. H. Maulud, S. R. M. Zeebaree, K. Jacksi, M. A. M. Sadeeq, and K. H. Sharif, “State of art for semantic analysis of natural language processing,” *Qubahan Academic Journal*, vol. 1, no. 2, pp. 21–28, Mar. 2021, doi: 10.48161/qaj.v1n2a44.

[19] G. Li *et al.*, “Research on the natural language recognition method based on cluster analysis using neural network,” *Mathematical Problems in Engineering*, vol. 2021, no. 2, pp. 1–13, May 2021, doi: 10.1155/2021/9982305.

[20] S. Meera and S. Geerthik, “Natural language processing,” in *Artificial Intelligent Techniques for Wireless Communication and Networking*, Wiley, 2022, pp. 139–153, doi: 10.1002/9781119821809.ch10.

[21] H. Yang and Y. Yang, “Design of english translation computer intelligent scoring system based on natural language processing,” *Journal of Physics: Conference Series*, vol. 1648, no. 2, p. 022084, Oct. 2020, doi: 10.1088/1742-6596/1648/2/022084.

[22] T. A. Koleck, C. Dreisbach, P. E. Bourne, and S. Bakken, “Natural language processing of symptoms documented in free-text narratives of electronic health records: a systematic review,” *Journal of the American Medical Informatics Association*, vol. 26, no. 4, pp. 364–379, Apr. 2019, doi: 10.1093/jamia/ocy173.

[23] Y. Cui, W. Che, T. Liu, B. Qin, S. Wang, and G. Hu, “Revisiting pre-trained models for Chinese natural language processing,” in *Findings of the Association for Computational Linguistics Findings of ACL: EMNLP 2020*, 2020, pp. 657–668, doi: 10.18653/v1/2020.findings-emnlp.58.

[24] T. C. Guetterman, T. Chang, M. DeJonckheere, T. Basu, E. Scruggs, and V. G. V. Vydiswaran, “Augmenting qualitative text analysis with natural language processing: methodological study,” *Journal of Medical Internet Research*, vol. 20, no. 6, p. e231, Jun. 2018, doi: 10.2196/JMIR.9702.

[25] F. Zhang, "Construction site accident analysis using text mining and natural language processing techniques," *Automation in Construction*, vol. 99, pp. 238–248, 2019, doi: 10.1016/j.autcon.2018.12.016.

[26] A. Jabbar, S. Iqbal, M. I. Tamimy, S. Hussain, and A. Akhunzada, "Empirical evaluation and study of text stemming algorithms," *Artificial Intelligence Review*, vol. 53, no. 8, pp. 5559–5588, Dec. 2020, doi: 10.1007/s10462-020-09828-3.

[27] F. Koto and I. Koto, "Towards computational linguistics in Minangkabau language: studies on sentiment analysis and machine translation," *arXiv:2009.09309*, 2020, doi: 10.48550/arXiv.2009.09309

[28] R. Regin, S. S. Rajest, and T. Shynu, "An automated conversation system using natural language processing (NLP) chatbot in python," *Central Asian Journal of Medical and Natural Science*, vol. 3, no. 4, pp. 314–336, 2022.

[29] K. Resiandi, Y. Murakami, and A. H. Nasution, "A neural network approach to create Minangkabau-Indonesia bilingual dictionary," *1st Annual Meeting of the ELRA/ISCA Special Interest Group on Under-Resourced Languages, SIGUL 2022 - held in conjunction with the International Conference on Language Resources and Evaluation, LREC 2022 - Proceedings*, pp. 122–128, 2022.

[30] H. Alshalabi, S. Tiun, N. Omar, F. N. AL-Aswadi, and K. Ali-Alezabi, "Arabic light-based stemmer using new rules," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 9, pp. 6635–6642, Oct. 2022, doi: 10.1016/j.jksuci.2021.08.017.

[31] M. N. Kassim, S. H. M. Jali, M. A. Maarof, and A. Zainal, "Towards stemming error reduction for Malay texts," in *Lecture Notes in Electrical Engineering*, vol. 481, 2019, pp. 13–23, doi: 10.1007/978-981-13-2622-6_2.

[32] A. P. Wibawa, F. A. Dwiyanto, I. A. E. Zaeni, R. K. Nurrohman, and A. N. Afandi, "Stemming javanese affix words using nazief and adriani modifications," *Jurnal Informatika*, vol. 14, no. 1, p. 36, 2020, doi: 10.26555/jifo.v14i1.a17106.

[33] A. Omar and W. I. Hamouda, "The effectiveness of stemming in the stylometric authorship attribution in Arabic," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 1, pp. 116–121, 2020, doi: 10.14569/ijacsa.2020.0110114.

[34] R. Pramana, Debora, J. J. Subroto, A. A. S. Gunawan, and Anderies, "Systematic literature review of stemming and lemmatization performance for sentence similarity," in *Proceedings of the 2022 IEEE 7th International Conference on Information Technology and Digital Applications, ICITDA 2022*, Nov. 2022, pp. 1–6, doi: 10.1109/ICITDA55840.2022.9971451.

[35] N. Swapna, P. Subhashini, and B. P. Rani, "Impact of stemming on telugu text classification," *International Journal of Recent Technology and Engineering*, vol. 8, no. 2, pp. 2767–2769, Jul. 2019, doi: 10.35940/ijrte.B2338.078219.

[36] D. Soyusiawaty, A. H. S. Jones, and N. L. Lestariw, "The stemming application on affixed Javanese words by using Nazief and Adriani algorithm," *IOP Conference Series: Materials Science and Engineering*, vol. 771, no. 1, Mar. 2020, doi: 10.1088/1757-899X/771/1/012026.

[37] M. S. Keezhatta, "Understanding EFL linguistic models through relationship between natural language processing and artificial intelligence applications.," *ERIC*, vol. 10, no. 4, pp. 251–262, Dec. 2019, doi: 10.24093/awej/vol10no4.19.

[38] M. Sarrouti and S. O. El-Alaoui, "SemBioNLQA: a semantic biomedical question answering system for retrieving exact and ideal answers to natural language questions," *Artificial Intelligence in Medicine*, vol. 102, p. 101767, Jan. 2020, doi: 10.1016/j.artmed.2019.101767.

[39] J. Singh and V. Gupta, "A novel unsupervised corpus-based stemming technique using lexicon and corpus statistics," *Knowledge-Based Systems*, vol. 180, pp. 147–162, Sep. 2019, doi: 10.1016/j.knosys.2019.05.025.

[40] R. Bekesh *et al.*, "Structural modeling of technical text analysis and synthesis processes," *CEUR Workshop Proceedings*, vol. 2604, pp. 562–589, 2020.

[41] A. Nzeyimana, "Morphological disambiguation from stemming data," in *COLING 2020 - 28th International Conference on Computational Linguistics, Proceedings of the Conference*, 2020, pp. 4649–4660, doi: 10.18653/v1/2020.coling-main.409.

[42] R. A. Ramadhani, I K. G. D. Putra, M. Sudarma and I. A. D. Giriantari, "Stemming algorithm for Indonesian signaling systems (SIBI)," vol. 5, no. 1, pp. 57–60, 2020, doi: 10.24843/IJEET.2020.v05.i01.p11.

[43] I. Mulyana, A. Suhendra, Ernastuti, and W. B. Agus, "Development of indonesian stemming algorithms through modification of grouping, sequencing and removing of affixes based on morphophonemic," *International Journal of Recent Technology and Engineering*, vol. 8, no. 2 Special Issue 7, pp. 179–184, Sep. 2019, doi: 10.35940/ijrte.B1044.0782S719.

[44] N. Yusliani, R. Primartha, and M. Diana, "Multiprocessing stemming: a case study of Indonesian stemming," *International Journal of Computer Applications*, vol. 182, no. 40, pp. 15–19, Feb. 2019, doi: 10.5120/ijca2019918476.

[45] F. S. Alotaibi and V. Gupta, "A cognitive inspired unsupervised language-independent text stemmer for information retrieval," *Cognitive Systems Research*, vol. 52, pp. 291–300, Dec. 2018, doi: 10.1016/j.cogsys.2018.07.003.

[46] N. Bölücü and B. Can, "Unsupervised joint PoS tagging and stemming for agglutinative languages," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 18, no. 3, pp. 1–21, Sep. 2019, doi: 10.1145/3292398.

[47] Y. Permana and A. Emarilis, "Stemming analysis Indonesian language news text with porter algorithm," in *Journal of Physics: Conference Series*, 2021, vol. 1845, no. 1, 12019, doi: 10.1088/1742-6596/1845/1/012019.

[48] F. W. Suci, N. Hayatin, and Y. Munarko, "In-Idris: modification of Idris stemming algorithm for Indonesian text," *IIUM Engineering Journal*, vol. 23, no. 1, pp. 82–94, Jan. 2022, doi: 10.31436/IIUMEJ.V23I1.1783.

[49] J. Jumadi, D. S. Maylawati, L. D. Pratiwi, and M. A. Ramdhani, "Comparison of Nazief-Adriani and paice-husk algorithm for Indonesian text stemming process," *IOP Conference Series: Materials Science and Engineering*, vol. 1098, no. 3, p. 032044, Mar. 2021, doi: 10.1088/1757-899x/1098/3/032044.

[50] N. A. Razmi, M. Z. Zamri, S. S. S. Ghazalli, and N. Seman, "Visualizing stemming techniques on online news articles text analytics," *Bulletin of Electrical Engineering and Informatics (BEEI)*, vol. 10, no. 1, pp. 365–365, Feb. 2021, doi: 10.11591/eei.v10i1.2504.

[51] L. Hickman, "Text preprocessing for text mining in organizational research: review and recommendations," *Organizational Research Methods*, vol. 25, no. 1, pp. 114–146, 2022, doi: 10.1177/1094428120971683.

[52] B. Garner, C. Thornton, A. L. Pawluk, R. M. Cortez, W. Johnston, and C. Ayala, "Utilizing text-mining to explore consumer happiness within tourism destinations," *Journal of Business Research*, vol. 139, pp. 1366–1377, 2022, doi: 10.1016/j.jbusres.2021.08.025.

[53] A.-I. Moreno and T. Caminero, "Application of text mining to the analysis of climate-related disclosures," *International Review of Financial Analysis*, vol. 83, 2022, doi: 10.1016/j.irfa.2022.102307.

[54] S. K. Tidke, P. V. Khedkar, A. Gupta, A. D. Goswami, M. Agrawal, and K. Jaggi, "Identification of chemical entities from prescribed drugs for ovarian cancer by text mining of medical records," in *2022 International Conference on Decision Aid Sciences and Applications, DASA 2022*, Mar. 2022, pp. 475–479, doi: 10.1109/DASA54658.2022.9765011.

[55] B. R. Lidiawaty, M. E. Zulfaqor, O. Diyantara, and D. R. S. Dewi, "Keywords generator from paragraph text using text mining in bahasa Indonesia," in *2022 International Conference on Interdisciplinary Research in Technology and Management, IRTM 2022 - Proceedings*, Feb. 2022, pp. 1–4, doi: 10.1109/IRTM54583.2022.9791753.

[56] N. B. Halvadia, S. Halvadia, and R. Purohit, “Using text mining to identify key dimensions of service quality for the Indian public sector banks’ mobile banking apps,” *Research Square*, 2022, doi: 10.21203/rs.3.rs-1536236/v1.

[57] A. Huang, Y. Zhang, J. Peng, and H. Chen, “Application of informetrics on financial network text mining based on affective computing,” *Information Processing and Management*, vol. 59, no. 2, Mar. 2022, doi: 10.1016/j.ipm.2021.102822.

[58] A. Y. Muaad *et al.*, “Arabic document classification: performance investigation of preprocessing and representation techniques,” *Mathematical Problems in Engineering*, vol. 2022, pp. 1–16, Apr. 2022, doi: 10.1155/2022/3720358.

[59] P. Willett, “The Porter stemming algorithm: then and now,” *Program*, vol. 40, no. 3, pp. 219–223, Jul. 2006, doi: 10.1108/00330330610681295.

[60] A. S. Rizki, A. Tjahyanto, and R. Trialih, “Comparison of stemming algorithms on Indonesian text processing,” *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 17, no. 1, pp. 95–102, Feb. 2019, doi: 10.12928/TELKOMNIKA.v17i1.10183.

[61] A. T. Ni’mah, D. A. Suryaningrum, and A. Z. Arifin, “Autonomy stemmer algorithm for legal and illegal affix detection use finite-state automata method,” *EPI International Journal of Engineering*, vol. 2, no. 1, pp. 46–55, Jun. 2019, doi: 10.25042/epi-ije.022019.09.

BIOGRAPHIES OF AUTHORS

Rini Sovia born in Solok, April 5, 1976. Lecturer at Putra Indonesia University YPTK Padang. She completed her education in informatics management in 1999 and master’s in informatics engineering in 2006. She is currently pursuing a doctorate in information technology. His areas of expertise include artificial intelligence (AI), expert systems (ES), data mining (DM), decision support systems (DSS), and databases. She can be contacted at email: rini_sovia@upiyptk.ac.id.

Prof. Dr. H. Sarjon Defit, S. Kom, MSc born in Padang Sibusuk 7 August 1970. He is the chancellor of Putra Indonesia University YPTK Padang. Currently active as a lecturer in computer science. The educational history of SI at the college of informatics and computer management (STMIK “YPTK” Padang) with a graduation in 1993. An educational history of S2 at Universiti Teknologi Malaysia, Johor Bahru, graduated in 1998. Then a doctoral education history at Universiti Teknologi Malaysia, Johor Bahru, graduated in 2003. The field of science consists of data mining, artificial intelligence, decision support systems, and others. He can be contacted at email: sarjon_defit@upiyptk.ac.id.

Yuhandri was born in Tanjung Alam on May 15, 1973. He is an assistant professor in Faculty of Computer Science, Universitas Putra Indonesia YPTK. He received the bachelor’s degree in informatics management and master’s degree in information technology in 1992 and 2006 from Universitas Putra Indonesia YPTK. Moreover, he completed his Doctor of Information Technology as informatics medical image expertise from Gunadarma University in April 2017. He is a lecturer at the Faculty of Computer Science, Universitas Putra Indonesia YPTK. He can be contacted at email: yuyu@upiyptk.ac.id.

Sulastri she obtained her master’s degree at Gadjah Mada University, Yogyakarta in 1997. In 2012 she obtained a doctorate degree at Padjadjaran University, Bandung. Since 1987 she has been a lecturer at the faculty of cultural sciences, Andalas University, Padang. Most of the research is in the field of cultural semiotics, and cultural studies analysis. She can be contacted at email: sulastri.sasindo@yahoo.com.