

E-Translator Kawi to Balinese

Oka Sudana^[1], Darma Putra^[2], M.Sudarma^[3], Rukmi S.Hartati^[4], I Putu Putra Diyastama^[5]
[1][2][5] Department of Information Technology and, Faculty of Engineering, Udayana University
[3][4] Department of Electrical Engineering, Faculty of Engineering, Udayana University

E-mail: agungokas@unud.ac.id

Abstract—Nowadays, technology development in cultural aspect has been rapidly evolved. The main reason for technological development in cultural aspect is to preserve the culture of previous era in this globalization age, particularly in Indonesia. Indonesia has many ethnics, races, languages, and cultures which have been passed from our ancestors, one of which is Kawi language. Kawi language is one of the languages which used in Javanese Hindu-Buddha Kingdoms. It was used in many literatures. At this time, only a few people understand Kawi language, particularly in Bali, so a preservation action must be done. Technology became a media to preserve and enhance understanding of Kawi language. Natural Language Processing like e-translator is the one that can be used to achieve the cultural preservation. E-translator application is a tool for translating one language to another. It translates Kawi to Balinese language word by word. Stemming methods used in this application are Bobby Nazief and Mima Adriani Algorithm, perfected by Kawi wording regulation. Translated results from Kawi to Balinese language can be in a form of a word, sentence, or paragraph. Quality measurement of the translation system has an accuracy of 82,27%. Hopefully, this application can be one of the media to preserve Kawi language in Bali and Indonesia.

Keyword - E-translator, Preservation, Culture, Kawi Language, Balinese language.

I. INTRODUCTION.

Preserving a culture is very vital in this modern era due to too many foreign cultures infiltrate Indonesia in this globalization age. Indonesia has many cultures, languages, ethnics, and tradition which must be protected. These days, many of the young generation begin to forget old cultures, languages, ethnics, and traditions. In this globalization age, technological development is very rapid, especially communication and language technology.

Preserving a language which is the main part of communication is very important so that culture and language inherited from the ancestors still can be exist in this modern era. Kawi language preservation is one of the examples. Kawi language preservation can be

done by construct a translator which is an application of Natural Language Processing. Natural Language Processing (NLP) is an Artificial Intelligence (AI) focused on processing natural language through web based Kawi to Balinese language translator application. Kawi to Balinese language translator application is considered necessary because there is not any electronic dictionary and the translation process was still using conventional dictionary, which takes too much time and hard to learn by people who do not understand Kawi language. Hopefully, this application can preserve one of the languages in Indonesia.

Translator application is an application which can be combined to a culture, especially language. Technology which combined to a culture has a very positive impact to language learning process. Language is a vital part of human life in manners and humanity itself.

II. RELATED WORK

There are many available translator applications and they can easily be found in the internet. Translator application can be made based on desktop, web, and mobile. There are some translator application made by using various methods, such as android based Bali to Indonesian language translator application that using binary search [1].

Another application such as dictionary and Indonesian to Balinese language translator using rule based method. Translation of Indonesian to Balinese language classified into 3 parts, including Bali Alus, Bali Madya, and Bali Kasar. The study was using research and development method with waterfall model [2], some researchers also write about implementation of Text to Speech application design for Balinese language. This study conclude that Text to Speech application for Balinese language which convert texts in Balinese language in Latin letters to speech with appropriate text pronunciation in Balinese language and save the speech into an output audio file with extension of *.wav and phoneme file with MBROLA input format with *.pho extension [3].

There is another study about language translation system, which is “Design and Implementation of English to Balinese Language Translation System with Rule Based Approach”. This English to Balinese language translation system is implemented using rule based approach with 4 work modules implementation,

including scanner, parser, translator, and evaluator module. Test and survey by 6 experts with 100 samples of English text in tourism field conclude that most experts stated that translation from the system are good enough and can be understood. It was proved with an average precision score of 89,3% [4].

Bali Alus language learning application development with PhoneGap was used as an offline learning media. This application was made by Waterfall method until the testing phase. This application test was done on Android platform with Blackbox Testing method. This application was made using HTML5, CSS, CSS JQuery Mobile framework, PhoneGap framework, and SQLite database [5].

Another study associated with language translation system is using Binary Search, which is a more efficient searching algorithm than Sequential Search. It is due to this algorithm does not need to explore every elements on the table. This study also tests the efficiency of sorting method such as Bubble sort, Selection sort, and Insertion Sort. The test result conclude that in order condition, time difference of Insertion Sort algorithm execution is 97% shorter than Selection Sort and Bubble sort algorithm, and in random condition, time difference of Insertion Sort algorithm execution is 84% shorter than Selection Sort and Bubble sort algorithm [6].

English to Indonesian language translation is using Rule Based Method, because the translation is done by bilingual (2 languages) which have different grammar. Utilization of Rule Based Method with Parsing Tree and production rule in English to Indonesian language translation can overcome the grammar difference from those 2 languages to make an appropriate translation [7].

There is another article about Android based Indonesian to Javanese language translation built design [8] Android based Indonesian to Javanese language translation built design [9] was made by a similar method with rule based and brute force method. Applications which use binary search method as a data search media and Nazief-Andriani algorithm as a processing method for words with affixes. This application showed an average translation time for 152,94 milliseconds per word in translation process, with an accuracy average of 73,33%.

A. Kawi Language

Kawi language is one of the language which existed in Java at time of Hindu-Buddha kingdoms and used in many literatures. Kawi language is also known as Jawa Kuna Language in Javanese Tradition, however Kawi language is not pure Jawa Kuna Language because it's already influenced by Sanskerta Language.

The name "kawi" itself was taken from "kayya" in Sanskerta Language, which means poem or verse. Kawi means a poet, a creator, or an author of classic literature. Based on this understanding, Kawi language means a language of an author or a language of poetry.

Kawi language has prefixes, suffixes, and infixes. These are some of the prefixes, suffixes, and infixes used as a reference in Kawi words.

TABLE I Prefix Kawi Language

No	Prefix Bahasa Kawi
1	a-
2	ma-
3	pi-
4	paha-
5	pa-
6	ka-
7	maka-
8	pinaka-
9	sa-
10	kapi-
11	kapa-
12	saka-
13	su-

TABLE II Sufix Kawi Language

No	Prefix Bahasa Kawi
1	-i
2	-aken
3	-an
4	-en

TABLE III Prefix Kawi Language

No	Prefix Bahasa Kawi
1	-in-
2	-um-

B. Balinese Language

Balinese language is an Austronesia language from Sundik branch, specifically from Bali-Sasak branch. It's mainly spoken in Bali island, west Lombok island, and in the east coast of Java island.

Balinese language has several levels in daily life that usually called *Sor-Singgih* Balinese Language. According to *Sor-Singgih*, Balinese Language composed of several parts such as word in *Alus Singgih* (ASI), word in *Alus Mider* (AMI), word in *Alus Sor* (ASO), word in *Mider/Andap*, and word in *Kepara* [1].

Word in *Alus Singgih* (ASI) means soft language that generally used to honor people with higher social status, beside that there is also word in *Alus Mider* (AMI) which can be used to honor or demean people according to their social status. Word in *Alus Mider* has 3 characteristics which can be used to honor, demean, and has a neutral aspect of words.

Then there is word in *Alus Sor* (ASO) which is a soft language that can be used to demean self and others. The fourth is word in *Mider* or words with neutral language. It means that *Mider* words have indifference sense of language, so it has no other using form. The fifth is word in *Andap* which has a neutral sense of

language, it's neither rude nor soft. Word in *Andap* can be used in conversation between good friends or relatives, and also when people from higher social status talk to the one from lower social status. Word in *Andap* is also known as polite rude words or honorable free words. The last one is word in *Kepara*, which are rude words. *Kepara* words are used in an unpleasant or emotional situation, so they are used in fights and curses.

C. Natural Language Processing

Natural language processing (NLP) is the second largest application after expert system in artificial intelligence field. The aim of this natural language is to enable making process of language computation model, so there will be interaction between human and computer with natural language media [10].

Basically, natural language is a representation form of a message that wants to be communicated among humans. The main form of its representation is voice/pronunciation, but it's not rare to be presented in the writing form. The point of natural language processing is disintegration of sentence or known as parser. Parser functions as a sentence reader, word by word and determine which kind of word follow the previous word. There are 2 kinds of application in this field which are text-based application and dialogue-based application. Text-based application includes all written text process application, such as books, news on newspaper, e-mails [11].

D. Tokenization

Tokenization is a process of breaking a stream of sentences into tokens. This is done by searching a space after each word. All the generated tokens are saved in a separate file for further processing. The list of tokens becomes input for further processing such as part-of-speech tagging and named entity recognition. Tokenization is the task of chopping given a character sequence and a defined document unit it up into pieces, called as tokens and discarding certain characters such as punctuation. It locates word boundaries. End point of a word and beginning of the next word is called as word boundary. It is also called as word segmentation [12].

In a concise form, tokenization is a process to break down texts in form of sentence, paragraph or document, into tokens or certain parts. The step to break down string into token is by eliminate punctuation marks and spaces, and then break down strings composed of sentences or paragraphs into words.

E. Stemming

Stemming is a process to look for root word. Stemming can eliminate morphology variation of words by deleting affixes of that word, so there will be a word with appropriate language morphology structure [13].

Stemming is a part of an information retrieval process. Information retrieval is a process to find information within a document, a process to find the document, a process to find data which explain the document, or finding process within a database including both stand-alone database relation and hypertext database. Information retrieval has 2 processes, they're indexing and searching. Indexing composed of 4 sub processes, including: Word token (alter document to a group of terms by eliminate all punctuation marks in that document and alter a group of terms into lowercase), Stop Word Removal (Deleting words process in the document such as: and, or, not, etc.), Stemming which is a process to alter a word into its root word, and Word Weighting which is a term weighting process in a document [14].

An example of stemming is when alter "bersama, kebersamaan, menyamai" into its root word, which is "sama". Stemming process in Indonesian language is different from stemming in English. Stemming process in English is only by eliminating the suffixes, while in Indonesian language, stemming process is eliminating suffixes, prefixes, and combining forms. Stemming algorithm for several languages have been developed, such as Porter algorithm for English, Porter algorithm for Indonesian language, Nazief and Adriani algorithm for Indonesian language.

F. Bobby Nazief and Mima Adriani Algorithm

This algorithm is based on Indonesian language morphology rule which grouping affixes into allowed and not allowed affixes [15]. This grouping includes front affix (prefix), end affix (suffix), middle affix (infix), and front and end affix combination (konfix) [16]. This Bobby Nazief and Mina Adriani Algorithm has several steps:

1. Find the word that will be stemmed in dictionary. If that word is found, then that word is assumed as a root word. Then the algorithm will stop.
2. Delete inflection suffixes ("-lah", "-kah", "-ku", "-mu", atau "-nya"). If it's a particle ("-lah", "-kah", "-tah", or "-pun"), then redo the step to delete possessive pronouns ("-ku", "-mu", atau "-nya").
3. Delete derivation suffixes ("-i", "-an", or "-kan"). If that word is found in the dictionary, then the algorithm will stop. If it's not found, then the algorithm will continue to step 3a
 - a. If "-an" has been deleted and the last letter of that word is "-k", then "-k" also be deleted. If that word is found in the dictionary, then the algorithm will stop. If it's not found, then the algorithm will continue to step 3b
 - b. Restore the deleted suffix ("-i", "-an", or "-kan") and continue to step 4.
4. Delete derivation prefix. If in step 3, there's a deleted suffix then continue to step 4a, if there isn't continue to step 4b.

- a. Check not allowed prefix-suffix combination in the table. If it's found, then the algorithm will stop. If it's not found, continue to step 4b..
 - b. For i, steps 1-3 are same, determine the prefix then delete it. If root word is still not found, continue to step 5, if it's found then the algorithm will stop and if the second prefix is same as the first one then the algorithm stop.
5. Recoding.
 6. If all process has been done but the root word is still not found, then the first word is assumed as a root word. Finish [17].

III. SYSTEM OVERVIEW

Web based Kawi to Balinese Language Translator application is an application using web interface which translate Kawi to Balinese Language. Web application is the one that can be accessed by web browser through internet or intranet connection. Web application is also a computer software coded in programming language which support web based software such as HTML, Java Script, CSS, PHP, etc.

This application is designed to work on web platform to translate sentences in Kawi Language to sentences in Balinese Language. Translator application is an application to help people who do not understand Kawi Language by translate it to Balinese Language so that they can understand.

The translator application is made for those who do not understand Kawi Language. Hopefully it can be a learning media to introduce Kawi Language in general, mainly to various places in Indonesia, especially in Bali. So the effort to preserve Kawi Language can be accomplished.

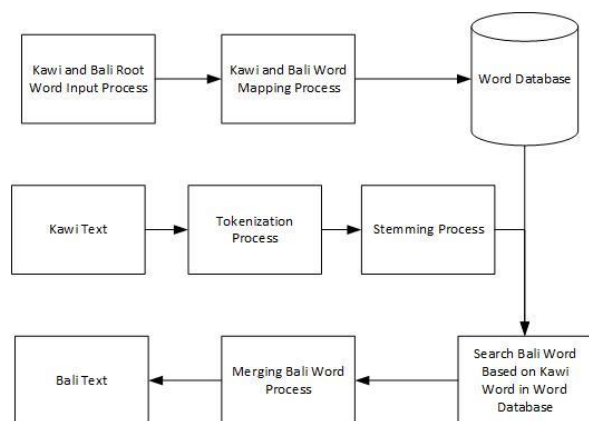


Fig. 1. System Overview

This application works by inputting Kawi Words or sentences in Latin letters. First, those Kawi words or sentences will undergo tokenization process to break down into some parts, and then it will continue to stemming process to get the root word in Kawi, so it will be easy to look for the associated word in database.

Previously, Kawi root words have been inputted into the database. The next step is mapping process from Kawi words to Balinese words. Kawi root words from the stemming process will be used to look for Balinese Words in the database. The result of the Bali Words from that searching process will be combined with other Bali Words, so the final result will be a text in Balinese Language.

This system has ERD that composed of 8 entities, and each of these entities has its own attribute which functioning as its own characteristic. These are the ERD of the web based Kawi to Balinese language translator application.

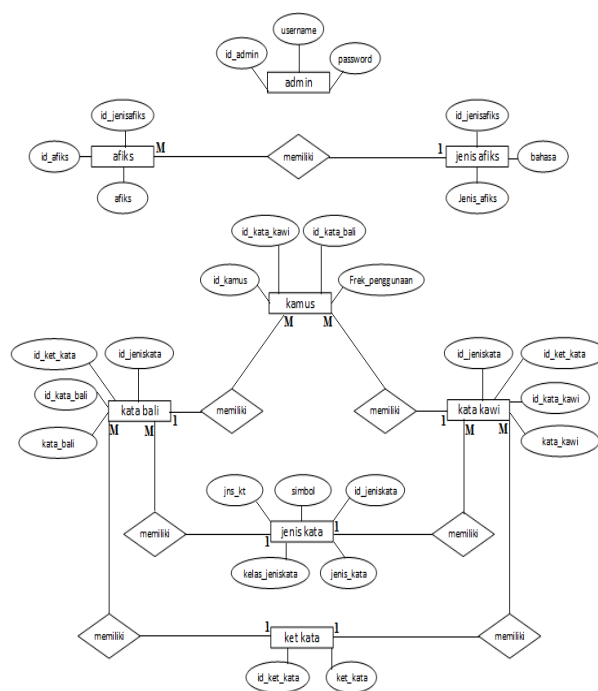


Fig. 2. Entity Relationship Diagram

Entities of ERD are Admin, Affix, Affix types, Word types, Balinese Words, Kawi Words, dictionary, and words. Each of that entity has its own attribute which functioning as its own characteristic.

IV. IMPLEMENTATION AND ANALYSIS

Kawi to Balinese language translator application is based on web. It composed of some user interfaces which used to translate Kawi to Balinese language text. This application interface view is very simple so people can use it easily and facilitate interaction with interface application. User starts to translate Kawi words or sentences by inputting them into the text area "Teks Bahasa Kawi" as shown in Fig 3.



Fig. 3. Kawi Language Text

Kawi text translated by some processes which are tokenization, stemming, word searching, and word combining. Tokenization is a process to break down texts in form of sentence, paragraph or document, into tokens or certain parts. This process is performed after text input process, and then that text will undergoes tokenization.

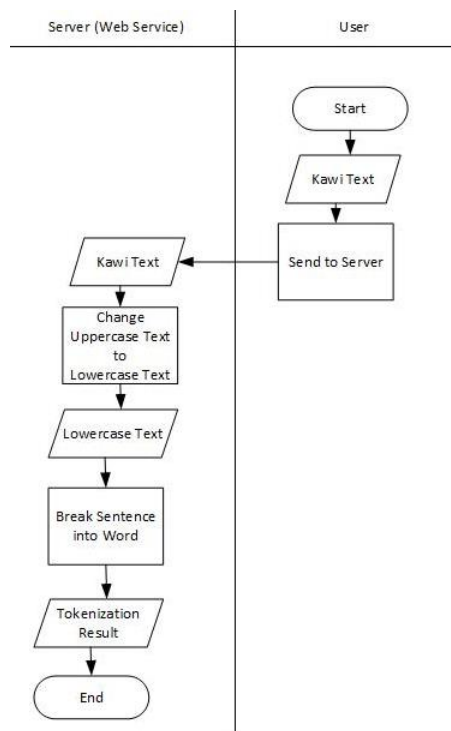


Fig. 4. Flowchart of Tokenization Process

Fig.4 is a tokenization process using Kawi language and producing words or phrases. The next process is stemming, where it eliminates morphology variations attached to a word by deleting affixes of that word, so there will be only a word with a correct Balinese language morphology structure. Stemming process will take the tokenized word, and then eliminate its prefixes, infixes, and suffixes to create its root word.

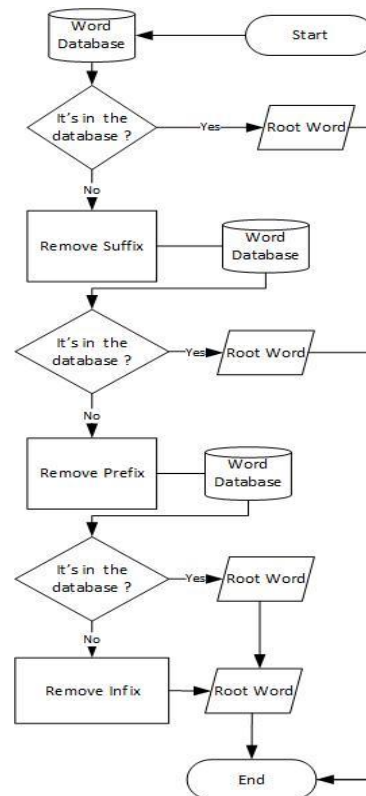


Fig. 5. Flowchart of Stemming Process

Stemming process in Fig. 5 is to find the root word of the input, after it get a result as shown in Fig. 6, the process continues to the next one which is searching the meaning of that root word. The result of the tokenization and stemming process of the input will be shown on system once it's done, as in Fig. 6.

Hasil Tokenisasi & Stemming		
10 =>	pranata =>	pranata
11 =>	jaya =>	jaya
12 =>	pandita =>	pandita
13 =>	ring =>	ring
14 =>	aji =>	aji
15 =>	kabeh =>	kabeh

Fig. 6. Stemming and Tokenization Result

Fig. 6 is the result of tokenization and stemming process. At first, the sentence will be broken down into words or phrases. That tokenized word will undergoes stemming process to find its root word. Stemming process will eliminate morphology variations attached to a word by deleting affixes such as prefix, infix, and suffix of that word, so the final result will be a word with correct morphology structure of Balinese language. Kawi language stemming process algorithm can be explained through these processes:

1. First, find the word that will be stemmed in root word dictionary, if it's found, then that word is assumed as a root word. The algorithm will stop if the root word has been found.
2. Delete these suffixes "-i", "-aken", "-an", "-en", dan "-ing" on the word, if the result word is found in the dictionary, then the algorithm will stop and if that result word is not found, then the algorithm will continue to c) to delete the prefix.
3. Delete prefixes "a-", "ma-", "pi-", "paha-", "pa-", "ka-", "maka-", "pinaka-", "sa-", "kapi-", "kapa-", dan "saka-" on the word, if the result word is found in the dictionary, then the algorithm will stop and if that result word is not found, then the algorithm will continue to d) to delete infix.
4. Delete infixes "-um-" and "-in-" on the word, if the result word is found in the dictionary, then the algorithm will stop.
5. All the steps are done, however if it has not work, then the first word is considered as a root word. Finish.

The next process is to find the translation of the word. Fig. 7 describes a process to find the translated word by looping until find the desired word. The translated word will enter the words combining process. Words combining is the next process after the translated word searching process, to combine that translated word into a sentence.

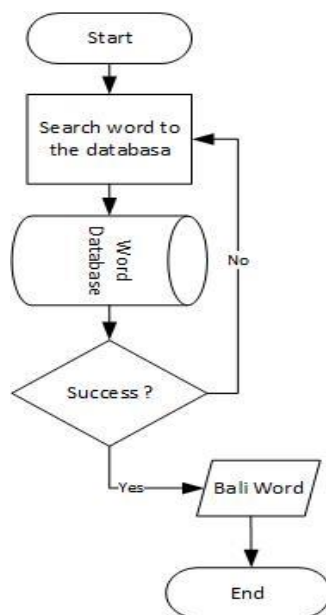


Fig. 7. Find Word of the Translation Flowchart

The result of combined words will be shown on Balinese language text result which its process is described in Fig. 8. Due to many translations of some processed words of Kawi to Balinese language, on the shown Balinese language result text, user can choose the appropriate translation.

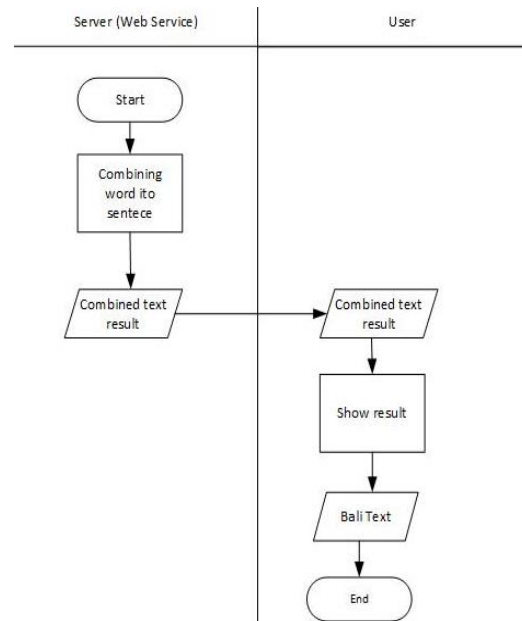


Fig. 8. Combined Words Flowchart

The result of the combined words will be shown on Balinese language text result as in Fig. 9. Some of the translations in Balinese language have more than 1 meaning, as the words in blue. The blue color means that it's a link used to select words. This selecting feature used to choose other words that considered more appropriate to the text translation.



Fig. 9 Balinese Language Text

User can choose the translation words by pressing the word if translation from the system is considered not appropriate. Word that has several translations will be shown in blue color. More often a translation is selected, then that translation will have a higher frequency. The higher frequency of a translation, then that word will show more often in text result.

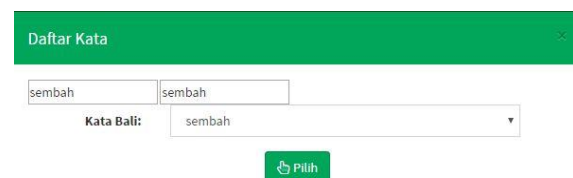


Fig. 10. Selecting other word translation

Fig. 10 describes an example of choosing other translation. Selecting other word translation can be used if there are more than one translation of Kawi word. Structure of the sentence from the translation process is still not very good, it needs some revision according to Bali language rule. Generally, sentence is a sequence of words with the appropriate rule. Word is

the smallest element of the language structure and sentence is the largest one [18].

However, this application is different with English to Bangla language, English to Balinese language, Indonesian to Madura language translator applications which using rule based method and save the basic rule of all knowledge regarding the problem. Kawi to Balinese language translator application does not use rule based method because it only fix the spelling of the result sentences. Translation time of Kawi to Indonesian language translator is shown in Table 4.

TABLE IV Prefix Kawi Language

No	Input Type	Time Process (ms)
1	Word	145.8
2	Phrase (2 word)	152.6
3	Sentence (5 word)	228.7
4	Paragraph (56 word)	824.5

Kawi to Balinese Language Translator can translate word, sentence, even paragraph with maximum of 400 words. Kawi to Balinese Language Translator is made to translate Kawi Language, which mostly used in kekawin. Kekawin sentence's structure is different from the structure of Balinese, Indonesian, or any other languages. It's because kekawin is a story delivered in rune or poem that can be sung, so in the process of translating there can be many translations and meanings, therefore this translator has a process to revise a sentence's structure once it's translated to Balinese language. It's different from android based Balinese to Indonesian translator application using binary search method, which translates Balinese to Indonesian language. The translated language, which is Balinese language, has a similar sentence's structure with Indonesian language so it's easy to translate each word and construct a good sentence [1].

V. CONCLUSION

Conclusion of this study is language translator can help people to study Kawi language in this modern age. This application works by inputting Kawi words or sentences in Latin letters. First, those Kawi words or sentences will undergo tokenization process to break down into some parts, and then it will continue to stemming process to get the root word in Kawi, so it will be easy to look for the associated word in database. Previously, Kawi root words have been inputted into the database. The next step is mapping process from Kawi words to Balinese words. Kawi root words from the stemming process will be used to look for Balinese words in the database. The result of the Bali Words from that searching process will be combined with other Bali words, so the final result will be a text in Balinese Language.

REFERENCES

[1] A. A. Kompiang Oka Sudana, I. K. A. Purnawan, dan N. M. R. M. Dewi, "Android based translator of balinese into indonesian using binary search method," in *International Journal of*

Software Engineering and its Applications, vol. 8, no. 6, 2014, hal. 165–182.

[2] I. M. G. Sunarya, "PENGEMBANGAN APLIKASI KAMUS DAN PENERJEMAH BAHASA INDONESIA – BAHASA BALI MENGGUNAKAN METODE RULE BASED BERBASIS ANDROID," in *Kumpulan Artikel Mahasiswa Pendidikan Teknik Informatika* vol. 4, hal. 70–81, 2015.

[3] P. Made, M. Pramadewi, M. Windu, dan A. Kesiman, "Pengembangan Aplikasi Text to Speech untuk Bahasa Bali," in *Jurnal Nasional Pendidikan Teknik Informatika* vol. 2, hal. 219–233, 2013.

[4] I Putu Deni Pratama, Agus Muliantara, "BAHASA BALI DENGAN MENGGUNAKAN PENDEKATAN BERBASIS ATURAN (RULE BASED)" in *Jurnal Ilmu Komputer* vol. 5, no. 1, hal. 47–54, 2012.

[5] N. Kadek dan A. Bintari, "APLIKASI PEMBELAJARAN BAHASA BALI ALUS DENGAN PHONEGAP (MELAJAH BASA BALI) BALINESE LANGUAGE LEARNING APPLICATION WITH PHONEGAP (MELAJAH BASA BALI)." in *e-Proceeding of Applied Science*, Vol. 3, no. 3, 2015

[6] D. M. Andayani, M. Yuliana, T. B. Santoso, L. Sinyal, P. Elektronika, dan N. Surabaya, "Penggunaan Metode Binary Search Pada Translator Bahasa Indonesia – Bahasa Jawa," hal. 1–8.

[7] E. Utami dan S. Hartati, "Pendekatan Metode Rule Based Dalam Mengalihbahasakan Teks Bahasa Inggris Ke Teks Bahasa Indonesia," *Jurnal Informatika*, vol. 8, hal. 42–53, 2007.

[8] F. Rohman, P. W. Buana, A. Agung, dan K. Wiranata, "Rancang Bangun Penerjemah Bahasa Indonesia ke Bahasa Jawa Berbasis Android," in *MERPATI*, vol. 3, no. 1, hal. 40–47, 2015.

[9] A. B. Setiawan, P. W. Buana, dan I. M. Sukarsa, "Aplikasi Translator Bahasa Jawa Ke Bahasa Indonesia Berbasis Android," in *MERPATI*, vol. 2, no. 3, hal. 344–350, 2014.

[10] T. C. Rindflesch, "Natural Language Processing," in *International Journal of Research in Engineering, Science and Technology*, vol. 2, no. 10, 2017.

[11] Preeti dan BrahmaleenKaurSidhu, "Natural language processing," *Int.J.Computer Technology & Applications*, vol. 4, no. 5, hal. 751–758, 2013.

[12] N. Munot dan S. Govilkar, "Rule Based Approach for Abstractive Text Summarization." in *International Journal of Research in Engineering, Science and Technologies*, vol. 1, no. 6, 2016.

[13] C. Moral, A. de Antonio, R. Imbert, dan J. Ram??rez, "A survey of stemming algorithms in information retrieval," *Information Research*, vol. 19, no. 1, hal. 76–80, 2014.

[14] M. R. Jayanthi dan M. C. Jeevitha, "An Approach for Effective Text Pre-Processing Using Improved Porters Stemming Algorithm," *IJISET -International Journal of Innovative Science, Engineering & Technology*, vol. 2, no. 7, hal. 797–807, 2015.

[15] J. Ring, "PENERAPAN ALGORITMA STEMMING NAZIEF & ADRIANI DAN SIMILARITY PADA PENERIMAAN JUDUL THESIS" in *Jurnal Ilmiah DASI*, vol. 15, no. 4, hal. 15–19.

[16] H. E. W. Mirna Adriani, Jelita Asian, Bobby Nazief, S. M.M. Tahaghoghi, "Stemming Indonesian: A confix-stripping approach," *ACM Transactions on Asian Language Information Processing*, vol. 6, no. 4, hal. 1–33, 2007.

[17] M. Adriani, J. Asian, B. Nazief, dan H. E. Williams, "Stemming Indonesian: A Confi x-Stripping Approach," *ACM Transactions on Asian Language Information Processing (TALIP)*, vol. 6, no. 4, hal. 1–33, 2007.

[18] P. N. Bali, "Perancangan Penganalisis Struktur Kalimat Bahasa Indonesia dengan Menggunakan Constraint-Based Formalism," in *Lontar Komputer*, vol. 5, no. 2, 2014.