Development of Indonesian Stemming Algorithms through Modification of Grouping, Sequencing and Removing of Affixes Based on Morphophonemic

Iyan Mulyana, Adang Suhendra, Ernastuti, Bheta Agus W

Abstract*: Text documents stored on the system in an unstructured form, so that the information inside cannot be extracted directly. To be able to extract it, it takes text processing which is first carried out initial processing (preprocessing text) to convert text documents into more structured by selecting words that used as indexes. The smaller the index value, the more text documents are recognized on the system and the information is more easily extracted. The size of the index determined by the number of groups of words formed. To avoid forming many groups of words, then each word is changed to become a basic word first before grouping. The process of changing of affix word into a basic word using certain rules is called stemming. This research aims to produce a new Indonesian stemming algorithm named UG18 Stemmer algorithm, which can reduce or eliminate stemming errors such as over-stemming and under-stemming on existing stemming algorithms including the Enhanced Confix Stripping (ECS) Stemmer algorithm and the New Enhanced Confix Stripping (NECS) stemming algorithm. The method used is the morphophonemic process approach, which sees affixes as bound morphemes that experience phoneme changes, phoneme addition, and phoneme removal. The three processes are mapped, and Finite State Automata was made to obtain new affixed groups, sequences and new deletion methods that form the basis of the development of the UG18 Stemmer algorithm. This algorithm developed not using a list of decapitation rules used in pre-existing algorithms. Decapitation rules replaced with morphophonemic based elimination rules. Based on the evaluation results and testing of the UG18 Stemmer algorithm, it has a lower error rate compared to the results of stemming using NESC Stemmer. The result can be seen from the randomized test of 2500 word using Relevance Judgment validated by Indonesian language experts, from 1.48% over-stemming and 16.69% under-stemming using the NECS stemmer algorithm down to 0.12% overstemming and 0% understemming using the UG18 algorithm stemmer. Also, the UG18 Stemmer algorithm can improve the speed performance process in the information retrieval-based document similarity measurement application of 45.47% compared to using the ECS stemmer algorithm.

***Index Terms*: Stemming, affixes, Morphophonemic, UG18 stemmer**

I. INTRODUCTION

In electronic document processing such as Information retrieval system and Document Similarity Measurement System for Plagiarism Examination, the text document stored in an unstructured form which is not stored in a table in a database. So that for ease of management, retention, and retrieval, each text document is stored in the form of a structured index. The index formed determines the process of document recognition in the system. The higher the index, the easier the text document will be recognized in the system. To form an index each document described in word form. Then each word is grouped according to the same word. Groups with the highest number of words will have the highest indexes. For example, the words makan, makanan, dimakan, memakan and termakan will be grouped into five groups, and each of them consists of one word. In this case will be different if the five words above omitted before they grouped. So that the same basic word obtained and only one group formed is the "makan" group with five words. Therefore before grouping, each word needs to be changed first into a basic word. The process of obtaining a basic word by removing affixes using certain rules is called stemming [1]. Some Indonesian language stemming algorithms based on basic word dictionaries have been developed including Nazief and Adriani algorithms in 2007 [2]. Confix Striping (CS) Stemmer developed by Asian J in 2007. Enhanced Confix Stripping (ECS) stemmer algorithm developed by Arifin in 2009 [3]. In 2016, Setiawan et al developed the stemming algorithm based on the classification of affix flexibility and the New Enhanced Confix Striping (NECS) Stemmer algorithm developed by Winarti [4]. The development of Indonesian stemmer shown in Fig.1.

One that affects the results of stemming is the grouping of affixes, sequences, and removing affixes. The affixes group is needed to establish the sequence of deletion. While the sequence and method of removal of affixes are very influential on the Stemming results. The five algorithms have similarities in particle grouping (lah, kah, tah, pun), pronouns (ku, mu, nya) and suffix (kan, an, i). Whereas the prefix of each algorithm has a different approach both in grouping and in its removal. The Nazief and

Revised Manuscript Received on April 25, 2019.

Iyan Mulyana, Department of Computer Science, Pakuan University, Bogor, Indonesia
Adang Suhendra, Department of Technique Informatic, Gunadarma University, Depok, Indonesia
Ernastuti, Faculty of Industrie Technology, Gunadarma University, Depok, Indonesia
Bheta Agus W, Department of Technique Informatic, Gunadarma University, Depok, Indonesia

Adriani stemming algorithm, CS Stemmer and ECS Stemmer classify the prefix into two groups, namely the default prefix "di", "ke", "se" and the prefix complex "me", "be", "te", "pe".

The default prefix is a prefix that has no changes, while the complex prefix is a prefix that will experience changes when met certain letters at the beginning of the word. For example, the prefix "me" will change to "mem" when met words with the initial letter "b", the prefix “me” will change to "meng" when met a word with "k". The complex prefix when deleted does not directly produce the basic word. To remove the complex prefix on Nazief and Adriani Stemmer's algorithm, CS Stemmer and ECS Stemmer is to use decapitation and deletion rules without ordering of deletion [3]. In the algorithm of affix classification flexibility-based and NECS stemmer algorithm, Prefix grouping was done based on the number of letters. For example two-letter prefixes ("di", "ke", "se", "me"), three-letter prefixes (ber, bel, ter, per), four-letter prefix ("meng", "meny", "peng", "peny") and so on. The way to eliminate the four-letter prefix on the algorithm based on affix classification and the NECS stemmer algorithm is to use several decapitations and deletion rules similar to ECS Stemmer [4].

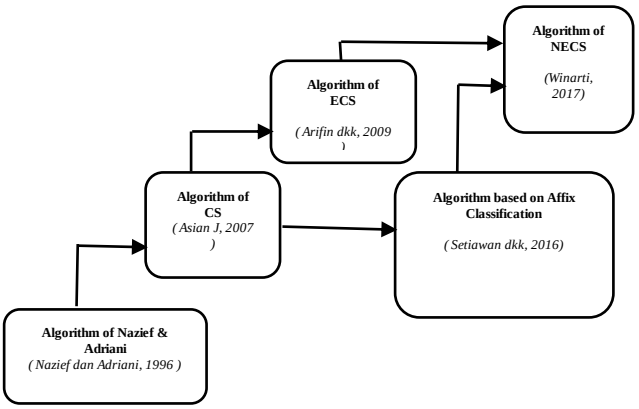

Fig. 1 The development of an Indonesian stemmer based on a basic word dictionary

The problems of the five algorithms above are in grouping prefixes, deletion sequences and deletion methods in the complex prefix. The absence of certain affixing groups has an impact on determining the sequence of elimination of affixes made. The inappropriate sequence of deletion results in over-stemming or under-stemming. The over-stemming condition is if too many parts of the word omitted. While under-stemming is if too few parts of the word omitted. [5]. In addition, the method of eliminating using decapitation rules has several disadvantages. There are still some rules that do not yet exist or are still wrong, resulting in less accurate stemming results. Besides, several decapitation rules are difficult to understand. So if it will be implemented into an application using a programming language, it will make a difference in implementing it and impacting on different stemming results.

One solution to solve some of the problems above is to develop a new algorithm by modifying the affixes group, sorting and removal method from the existing stemming algorithm through morphophonemic approaches. In this approach, affixes are seen as bound morphemes which will experience changes, additions and phoneme removal [6]. Then it is mapped to produce groups, sequences and removing new affixes. The morphophonemic process is the process of phonemic changes that arise in word formation due to the meeting of morphemes with other morphemes [7]. Also, one alternative that can be done to overcome the mistakes that often occur in stemming processes can influence the correctness such as over-stemming and under-stemming is to set the order of deletion of each group of affixes formed. Then it was followed by deletion rules for prefixes that experienced phoneme changes, phoneme addition, and phoneme removal. So there is no need to use decapitation rules.

II. METHODOLOGY

The general description of the research is to develop a new Stemming Algorithm, named the UG18 Stemmer Algorithm as shown in Fig. 2. This algorithm developed by modifying the affixes group, sorting and method for removing the affixes from the existing algorithm, ECS Stemmer and NECS Stemmer algorithm. The UG18 Stemmer algorithm was developed to correct some of the weaknesses found in the algorithm. The general description of the development of the Stemmer UG18 Algorithm shown in Fig. 3.

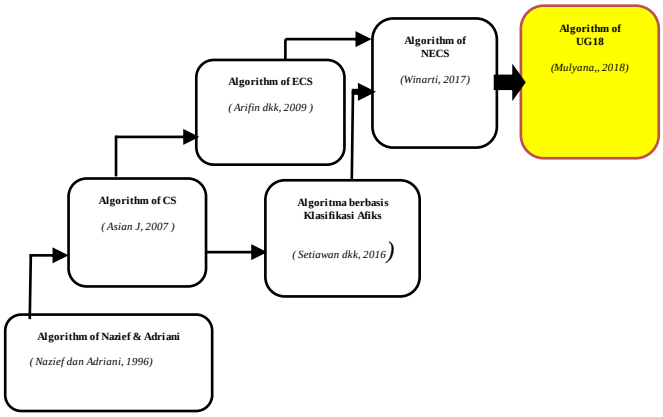

Fig. 2. General Description of Research

The carried out with a morphophonemic approach by placing affixes as bound morphemes especially for complex prefixes namely me, pe, be and te. This is because the bound morpheme if met with another morpheme, will experience phonemic changes called morphophonemic processes. There are three morphophonemic processes, morphophonemic in the form of phoneme changes, phoneme addition and phoneme removal [7]. The results of this morphophonemic process are in the form of affix mapping that forms the basis for creating new affixed groups, sorting and methods for eliminating new affixes so that it is easier to develop the UG 18 Stemmer algorithm..

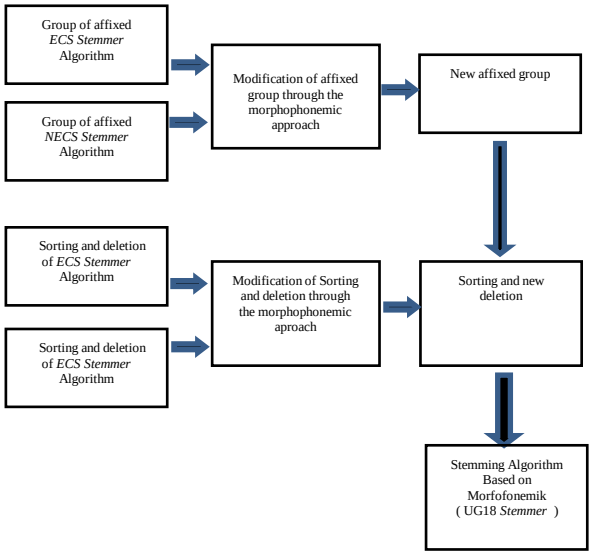

Fig. 3 General description of the development of the Stemmer UG18 algorithm

A. Creating the Affixes group of Stemming Algorithms Through the Morphophonemic Approach (UG18 Stemmer)

In this study, the creating of affixes groups was carried out through a morphophonemic approach called affixes perceived as bound morphemes. In general, there are two kinds of bound morphemes namely morphemes without morphophonemic processes and morphemes with morphophonemic processes. Morphemes without morphophonemic processes include particle morphemes, pronoun morphemes, suffix morphemes, infix morphemes and prefix morphemes without morphophonemic processes (di, ke and se). While morphemes with morphophonemic processes are prefixed morphemes (me, pe, be and te). Besides that, based on the solution to solve the problem as shown in table 3.10, the combined prefix such as memper, mempel, menter, diper, dipel and diter are added. The affixes groups developed are shown in Table 1.

Table 1. Affixed groups on stemming algorithms through the Morphophonemic approach (UG18 Stemmer)

No	Affixed Group
1	Prefix without Morphophonemic Process (‘di-‘, ‘ Ke-‘, ‘ Se-‘)
2	Prefix without Morphophonemic Process (‘-me-‘, ‘be-‘, ‘pe-‘, ‘te-‘)
3	Combined Prefix (‘Memper-‘, ‘Mempel-‘, ‘Menber-‘)
4	
5	Suffix (‘-kan‘,‘-an‘,‘-i‘)
	Possessive Pronouns (‘-Ku‘,‘-Mu‘,‘ -Nya‘)
7	Particle ‘-Lah‘,‘-Kah‘,‘ -Tah‘,‘- Pun‘)

B. Development of Affixes Removal in Stemming Algorithms through Morphophonemic Approach (UG18 Stemmer)

In this study, sorting and removing methods based on the analysis results of the ECS Stemmer and NECS Stemmer algorithms as well as through the morphophonemic approach. All morphemes without morphophonemic processes add by the combined prefix morpheme removed first to prefixing with the morphophonemic process. Except for the suffix morphemes whose deletions are combined with the initial morpheme with the morphophonemic process. The removing stages are generally shown in Fig. 4.

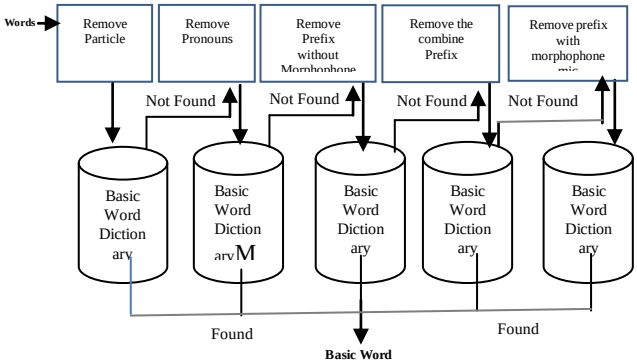

Fig. 4 Stages of Affixes Removal in the UG18 Stemmer Algorithm

C. Development of Removal Methods for the UG18 Stemmer Algorithm

In this study, the method of elimination for all morphemes without morphophonemic processes including the combined prefix morpheme was carried out by directly removing the morpheme. Whereas to start with the morphophonemic process the method of removal cannot be done directly. This is because several phonemes experience changes due to the encounter of one morpheme with another morpheme. In the study developed a method of elimination without using decapitation rules carried out morphophonemic process approach that is initiated by mapping how the morphophonemic process in the "me", "pe", "be" and "te" processes. There are three forms of the morphophonemic process which are analyzed, morphophonemic process in the form of phoneme changes, phoneme addition, and phoneme removal. The next stage is a summary of the morphophonemic process for the prefixes me, pe, be and te. Then as the final step is to create stages for deleting the prefixes me, pe, be and te as the basis for making the algorithm. The stages are as follows:

a. Mapping the Morphophonemic Process in the Form of Phoneme Change, Addition of Phonemes and Phoneme Removal.
b. Summarize the morphophonemic process for the prefixes me, pe, be and te based on phoneme removal, phoneme changes, and phoneme addition. As shown in Table 2.
c. Make stages of the affixes removal method based on morphophonemic processes for the prefix me, pe, be and te. As shown in Table 3 to Table 6

Table 2 Summary of the Morphophonemic Process in the prefixes me, pe, be and te

Phoneme	Phoneme Removal	Phoneme changes	Phoneme Addition
Me*N*	Me	Meng Mem Men Meny	Menge
Pen*N*	Pe	Peng Pem Pen Peny Per Pel	Penge
Be(r)	Be	Ber Bel	-
Pe(r)	Pe	Per Pel	-
Te(r)	Te	Ter	-

Table 3 Stages of deletion in the "me" prefix

Words	First deletion	Second deletion
Prefix “Menge”	Prefix “me”	Prefix “nge”
Prefix “Meng”	Prefix “me”	Prefix “ng”
Prefix “Mem”	Prefix “me”	Prefix “m”
Prefix “Men”	Prefix “me”	Prefix “n”
Prefix “Meny”	Prefix “me”	Prefix “ny”

Table 4. Stages of deletion in the “pe” prefix

Words	First deletion	Second deletion
Prefix “Penge”	Prefix “pe”	Prefix “nge”
Prefix “Peng”	Prefix “pe”	Prefix “ng”
Prefix “Pem”	Prefix “pe”	Prefix “m”
Prefix “Pen”	Prefix “pe”	Prefix “n”
Prefix “Peny”	Prefix “pe”	Prefix “ny”
Prefix “Per”	Prefix “pe”	Prefix “r”
Prefix “Pel”	Prefix “pe”	Prefix “l”

Table 5 Stages of deletion in the “be” prefix

Words	First deletion	Second deletion
Prefix “Ber”	Prefix “be”	Prefix “r”
Prefix “Ber”	Prefix “be”	Prefix “l”

Table 6. Stages of deletion in the “te” prefix

Words	First deletion	Second deletion
Prefix “ter”	Prefix “te”	Prefix “r”

III. RESULTS AND DISCUSSION

The UG18 Stemmer algorithm, as well as the ECS Stemmer algorithm and NECS stemmer, are stemming based dictionaries. The UG18 Stemmer algorithm is a modification of the ECS Stemmer algorithm and NECS stemmer through a morphophonemic approach. The morphophonemic approach is carried out by placing affixes as bound morphemes, especially for complex prefixes, namely ‘me‘, ‘pe-‘, ‘be-‘ and ‘te-‘. This is because the bound morpheme if met with another morpheme will experience phonemic changes called morphophonemic processes. There are three morphophonemic forms, morphophonemic in the form of phoneme changes, phoneme addition and phoneme removal [7]. The results of the morphophonemic process are in the form of affix mapping which forms the basis for creating new affixes groups, sequences and methods for eliminating new affixes, making it easier to develop the UG18 Stemmer algorithm. According to the affixes group, the sequencing and the removal method that has been produced are shown in Figure 5 to Figure 9.

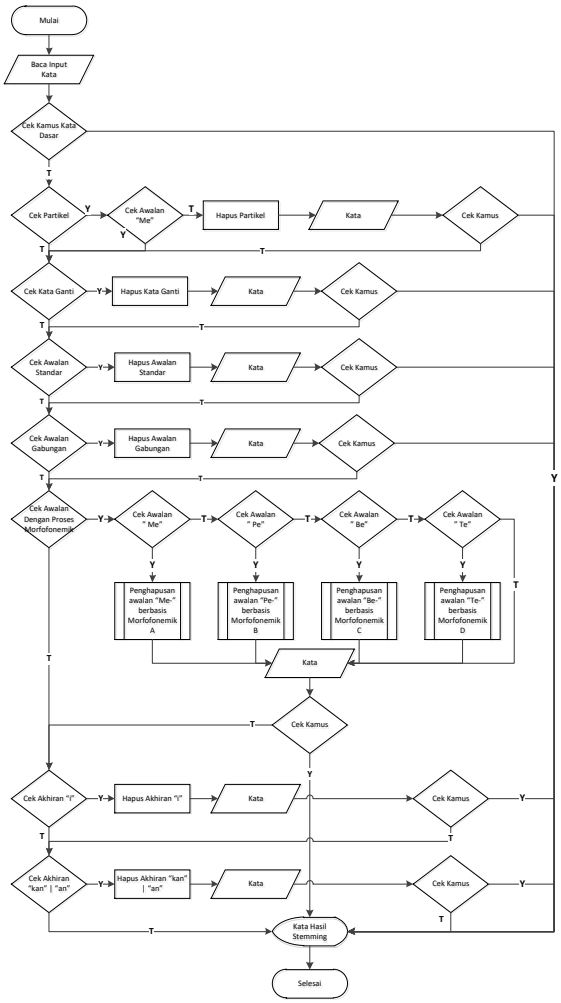

Fig. 5 Flowchart of the UG18 Stemmer Algorithm

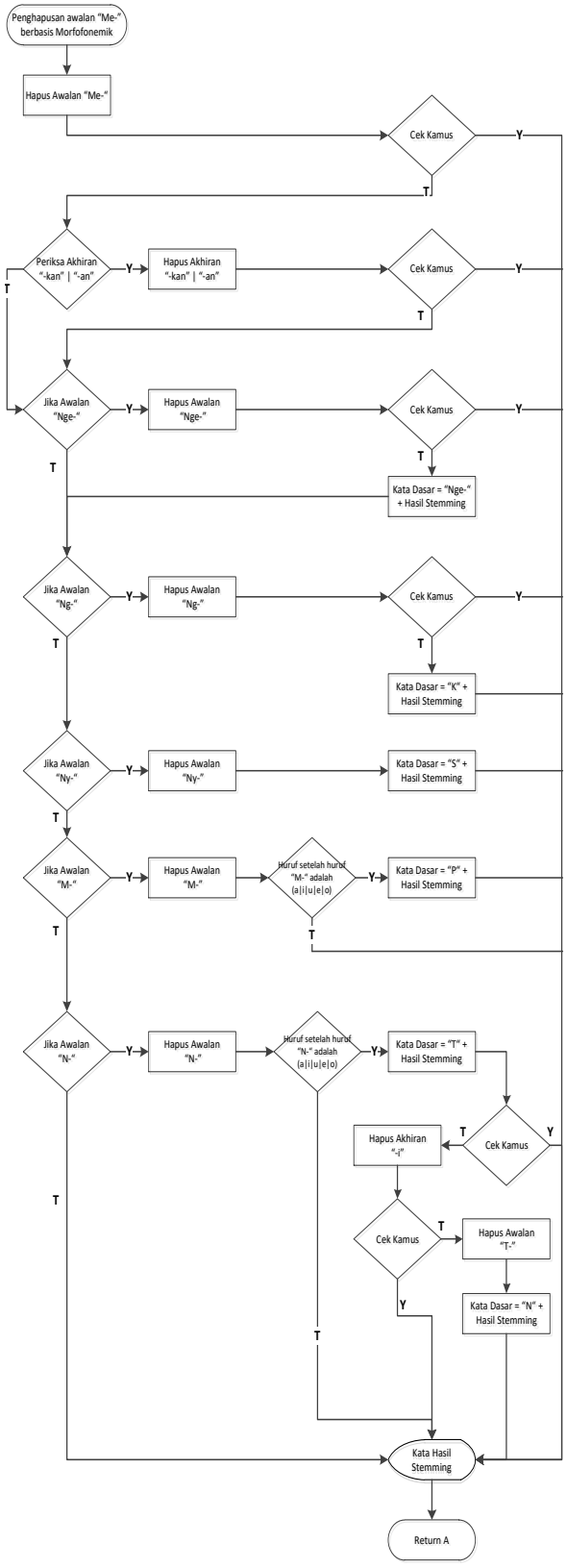

Fig. 6. UG18 Stemmer algorithm flowchart for the prefix 'me'

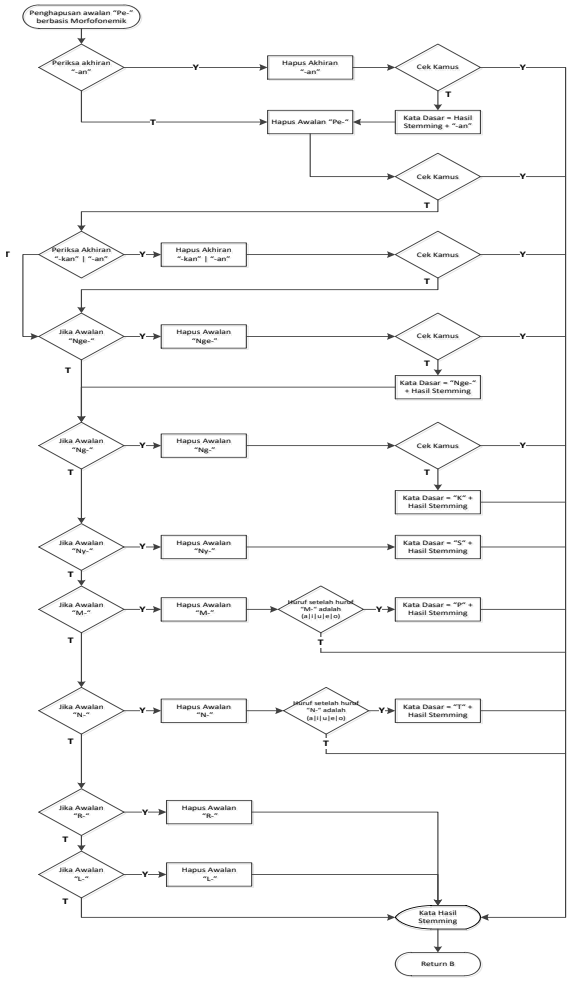

Fig. 7 UG18 Stemmer algorithm flowchart for the prefix 'pe'

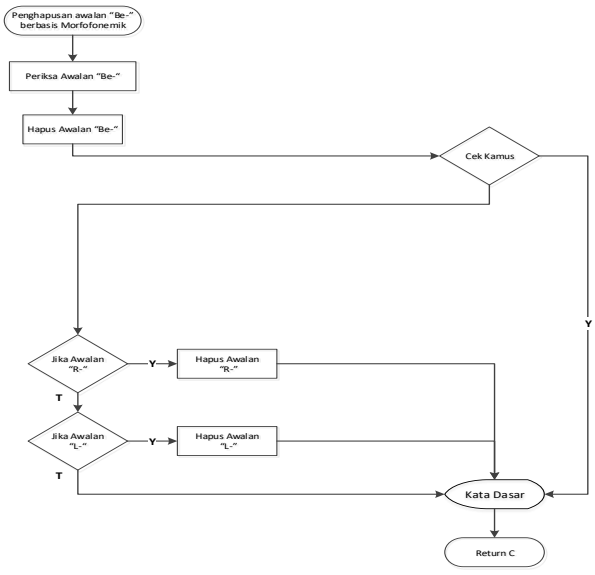

Fig. 8. Flowchart of the UG18 Stemmer algorithm for the prefix 'be'

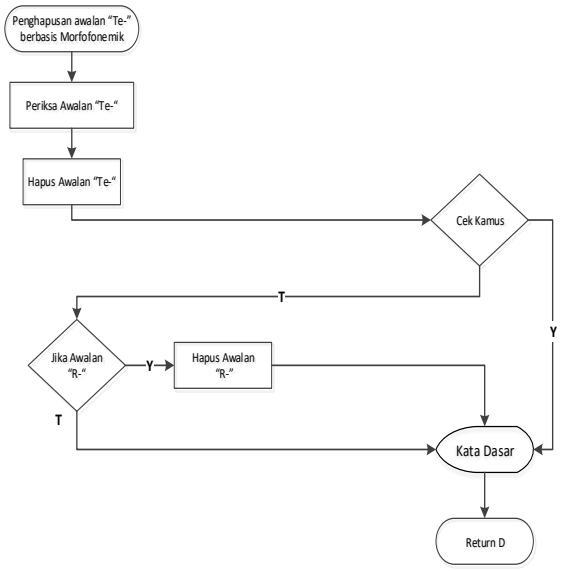

Fig. 9 Flowchart of the UG18 Stemmer algorithm for the prefix 'te'

To evaluate and to test the morphophonemic-based stemming algorithm (UG18 Stemmer), a stemming application was made using UG18 Stemmer. The application was made to test stemming results per word and per document — the appearance of the application for testing stemming results per word is shown in Fig.7 and Fig. In this application, stemming affix words or repeated words can be done to find the basic words. In the process, the affixed word entered into the application; then the affixing removal process will be carried out. Then the stemming results are checked in the dictionary database; whether the basic word found. If it found, then the word sought considered as a basic word.

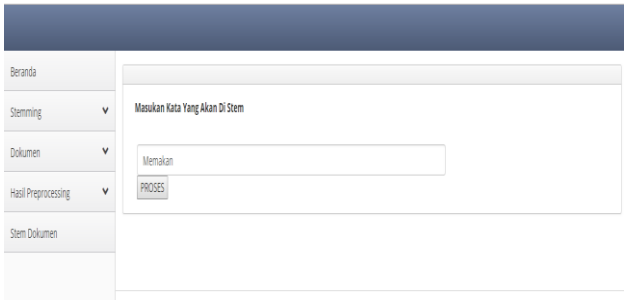

Fig. 7 Display of Input for affix words

Fig. 8 Output display of affix word

After evaluating stemming results per word using the UG18 stemmer algorithm application as shown in Fig. 6 and Fig. 7 of the 500 words in randomly, generated stemming results as shown in Table 7.

Table 7. Test results of the correctness using the NECS Stemmer and UG18 Stemmer algorithms

Number of words	The result of the *ECS Stemmer*				The result of the *UG18 Stemmer*			
	Overstemming		*Onderstemming*		*Overstemming*		*Understemming*	
2500	37	1,48 %	424	16,96 %	3	0,12 %	0	0 %

In Table 7, it can be seen that the results of the stemming process using the UG18 Stemmer algorithm can reduce the level of Under-stemming found in stemming results using NECS stemmer. Whereas for over-stemming both using the UG18 Stemmer and NECS Stemmer have the same amount is five words. from the five words that are still over-stemming, there is the incorrect result, but from the Relevance-Judgment assessment is correct because there are in the basic words.

IV. CONCLUSION

1. Morphophonemic based stemming algorithm (UG18 Stemmer) is a stemming algorithm that is developed without using decapitation rules but has a high correctness level. The result can be seen from the randomized test of 2500 word using Relevance Judgment validated by Indonesian language experts, from 1.48% over-stemming and 16.69% under-stemming using the NECS stemmer algorithm down to 0.12% over-stemming and 0% using the UG18 algorithm stemmer. Also, the UG18 Stemmer algorithm can improve the speed performance process in the information retrieval-based document similarity

2. Morphophonemic-based stemming algorithms (UG18 Stemmer) have a fairly low error rate both oversteeming and understemming. The stemming algorithm can be seen from the stemming trial using a stemming application that uses the Stemmer UG18 algorithm

REFERENCES

1. R. Setiawan, A. Kurniawan, W. Budiharto, I. H. Kartowisastro and H. Prabowo. "Flexible affix classification for stemming Indonesian Language." In 2016 13th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, pp. 1-6, 2016.
2. M. Adriani, J. Asian, B. Nazief, S. M. Tahaghoghi and H. E. Williams. "Stemming Indonesian: A confix-stripping approach." ACM Transactions on Asian Language Information Processing, vol. 6, no. 4, pp. 1-33, 2007.
3. A. Z. Arifin, I. Mahendra and H. T. Ciptaningtyas. "Enhanced Confix Stripping Stemmer and Ants Algorithm for Classifying news Document in Indonesian Language." In Proceeding of International Conference in Information 7 Communication Technology and System, pp. 149-157, 2009.
4. T. Winarti, J. Kerami ans S. Arief. "Determining Term on Text Document Clustering using Algorithm of Enhanced Confix Stripping Stemming." International Journal of Computer Applications, vol. 157, no. 9, pp. 8-13, 2017.
5. B. Zaman. "Modifikasi Algoritma Porter untuk Stemming pada Kata Bahasa Indonesia." In Proseding Seminar Nasional Teknologi Informasi dan komunikasi, pp. 543-550, 2014.
6. A. Mulyadi and A. M. Fajwah. Intisari Tata Bahasa Indonesia. Bandung: Yrama Widya
7. Sugerman. Morfologi Bahasa Indonesia, Kajian Kearah Linguistik Deskriptif. Yogyakarta: Penerbit Ombak, 2016.