IJCCS (Indonesian Journal of Computing and Cybernetics Systems)
Vol.13, No.4, October 2019, pp. 379~388
ISSN (print): 1978-1520, ISSN (online): 2460-7258
DOI: https://doi.org/10.22146/ijccs.49072

Modification of Stemming Algorithm Using A Non Deterministic Approach To Indonesian Text

Wafda Adita Rifai*[1], Edi Winarko[2]
[1]Master Program of Computer Science, FMIPA UGM, Yogyakarta
[2]Department of Computer Science and Electronics, FMIPA UGM, Yogyakarta
e-mail: ***[1]wafda.a@mail.ugm.ac.id** , [2]ewinarko@ugm.ac.id

Abstrak

Dalam ilmu Artificial Intelligence terdapat bidang ilmu yang secara spesifik berfokus pada pengolahan bahasa yaitu Natural Language Processing (NLP). Salah satu tahapan yang dilakukan pada NLP adalah Preprocessing. Preprocessing merupakan tahapan dalam menyiapkan data sebelum diolah. Salah satu tahap pada proses preprocessing adalah Stemming. Stemming merupakan proses yang melakukan pencarian kata dasar dari suatu kata. Pemilihan kata dasar yang tidak tepat dapat menjadi kesalahan informasi yang akan diperoleh. Selain itu, proses stemming tidak selalu menghasilkan satu kata dasar karena terdapat beberapa kata dalam Bahasa Indonesia yang memiliki dua kemungkinan, yaitu sebagai kata dasar maupun kata berimbuhan seperti pada kata "beruang"

Penelitian ini melakukan modifikasi pada salah satu metode Stemming yang ada dengan menerapkan pendekatan non deteministik untuk meningkatkan akurasi. Penggunaan metode non deterministik dilakukan dengan menerapkan daftar kandidat kata dasar untuk kata yang memiliki kata dasar lebih dari satu. Dari daftar kandidat kata dasar tersebut kemudian dipilih salah satu kata sebagai hasil.

Modifikasi stemming ini telah diuji pada data sejumlah 15.934 dengan tingkat akurasi sebesar 93%. Oleh karena itu modifikasi stemming ini dapat digunakan untuk mengidentifikasi kata yang memiliki kata dasar lebih dari satu.

Kata kunci— *stemming, non deterministik, akurasi*

Abstract

Natural Language Processing is part of Artificial Intelegence that focus on language processing. One of stage in Natural Language Processing is Preprocessing. Preprocessing is the stage to prepare data before it is processed. There are many types of proccess in preprocessing, one of them is stemming. Stemming is process to find the root word from regular word. Errors when determining root words can cause misinformation. In addition, stemming process does not always produce one root word because there are several words in Indonesian that have two possibilities as root word or affixes word, e.g.the word "beruang".

To handle these problems, this study proposes a stemmer with more accurate word results by employing a non deterministic algorithm which gives more than one word candidate result. All rules are checked and the word results are kept in a candidate list. In case there are several word candidates were found, then one result will be chosen.

This stemmer has been tested to 15.934 word and results in an accurate level of 93%. Therefore the stemmer can be used to detect words with more than one root word.

Keywords*—stemming, non deterministik, accurate*

Received August 28th,2019; Revised October 17th, 2019; Accepted October 21th, 2019

1. INTRODUCTION

One of the key innovations needed to achieve the desired level of artificial intelligence is a machine that can process and interpret textual data. One of the fields in artificial intelligence science specifically focusing on language processing is called natural language processing (NLP). The stages in NLP include pre-processing. Pre-processing is crucial to the entire field as it deals with the preparation of data before the processing stage begins. If researchers were to skip this stage, it would cause the data to appear inconsistent, and this may result in an unfavorable outcome.

Stemming is a stage in text processing aiming to find the root word of an original word that appears in a text. The stemming stage is often used in text-based applications including search engines, machine language translators, chatbots and spelling checkers. There are two types of stemming methods for Bahasa, including dictionary-based and rule-based [1,2,3] algorithms.

Reviewed from accuracy point, dictionary-based algorithm is actually better than the rule-based one [4,5,6]. One of the most used dictionary-based algorithms for stemming is Confix Stripping Stemmer [7]. This algorithm for stemming was then developed into Enhanced Confix Stripping Stemmer [8] to result in improved accuracy.

Although the development of algorithms for stemming has always been encouraged, problems are still inevitable, including overstemming, where words are cut excessively after stemming, and understemming, where there is not enough cutting of words after stemming is completed. To avoid overstemming, most stemming algorithms depend on the completeness of the dictionary to check whether the root word of a word can be found, given that it has been through stemming process. This would later cause another problem to occur, which is the dependency on comprehensive dictionary. Additionally, Word-Sense Ambiguity may occur, where one word can have more than one meaning and one word can have more than one root word, such as the word “beruang” in Bahasa having “beruang” and “uang” as its root words. The existing algorithms for stemming today cease to identify the possibility of a word having two root words, for the algorithms would just stop once the root word of a word has been found and the output would only be based on the findings from the dictionary.

Problems of double root words have previously been handled using the non-deterministic approach[9, 10]. Non-deterministic algorithms can provide different outputs for the same input at different executions. Unlike deterministic algorithms that only provide one output for the same input even on different operations. Non-deterministic algorithms are useful for finding approximate solutions when the right solution is hard to derive using deterministic algorithms. The implementation of the non deterministic method into stemming process was conducted by putting out some possibilities of root words found in a word into a list of root word candidates. The correct selection of root word candidates is a challenge by itself[11].

Based on this background, the research aims to identify the possibility of forming more than one root word in a word. The data used include translations of the Qur'an and Sherlock Holmes novels.

2. METHODS

2. 1 System Analysis

This research uses Enhanced Confix Stripping Stemmer method, which has been modified. The modification was completed by implementing the concept of non-deterministic Enhanced Confix Stripping algorithm. The concept itself is about putting root words found from each word in the data to a list of candidates. When a root word is found in the dictionary, the algorithm does not stop. It will then be recorded in the list of candidates, and the next steps of Enhanced Confix Stripping Stemmer method will follow. The last stage of the process will be about selecting the root words that are going to be presented as the result.

2.2 Data Collection

The data used in this research include the translation of the Qur’an (in Bahasa) and Sherlock Homes novel (25.951 words). Before this data is used, pre-processing must be completed. Pre-processing includes cleaning and tokenization. After pre-processing, 15.934 unique words are found.

Stages of pre-processing include :

Cleaning

At this stage, data cleaning is conducted to remove unnecessary characters, numbers, and symbols unread in the text (example: €, □, æ, ǽ). The following is the example of sentences executed in the cleaning process: “Pergilah diwaktu pagi (ini) ke kebunmu jika kamu hendak memetik buahnya” after the cleaning process is completed, the sentences become: Pergilah diwaktu pagi ini ke kebunmu jika kamu hendak memerik buahnya.

Tokenization

At this stage, the words are cut accordingly to the order decided beforehand. If there is any repetition/identical word found in a sentence structure, it will still be cut accordingly. For example: Pergilah diwaktu pagi ini ke kebunmu jika kamu hendak memetik buahnya This *tokenization* stage separates the words. The following is the example, separated by | sign into: Pergilah | diwaktu | pagi | ini | ke | kebunmu | jika | kamu | hendak | memetik | buahnya

2. 3 System Architecture

Process flow comparisons of Enhanced Confix Striping and modified Enhanced Confix Striping using non deterministic method as presented in Figure 1.

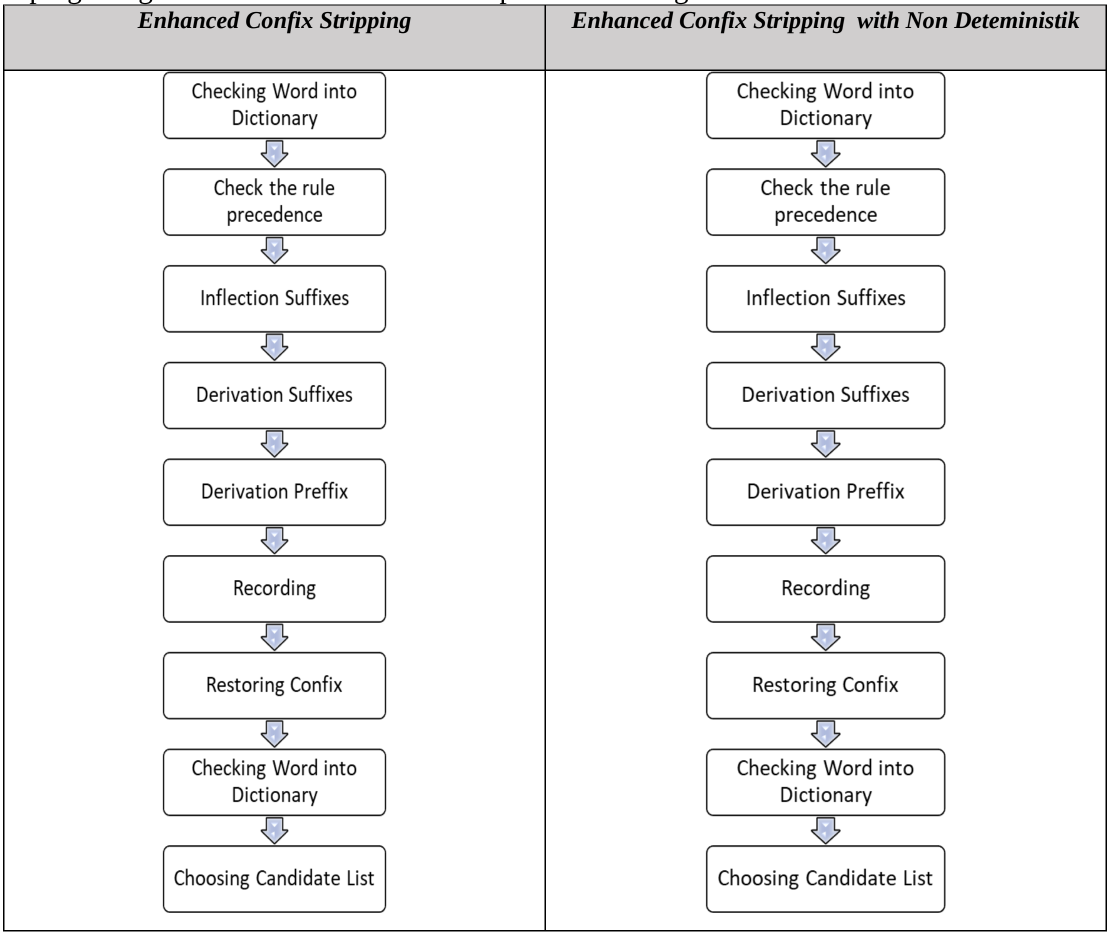
Enhanced Confix Stripping
Enhanced Confix Stripping with Non Deteministik
Checking Word into Dictionary
Check the rule precedence
Inflection Suffixes
Derivation Suffixes
Derivation Preffix
Recording
Restoring Confix
Checking Word into Dictionary
Choosing Candidate List
Checking Word into Dictionary
Check the rule precedence
Inflection Suffixes
Derivation Suffixes
Derivation Preffix
Recording
Restoring Confix
Checking Word into Dictionary
Choosing Candidate List

Figure 1 Flowchart of Enhanced Confix Striping using non deterministic method

The following is an explanation of the system architecture in Figure 1:

1. Checking Word into Dictionary
   In this step, if the word is found in root word dictionary then the word will be saved as candidate root word.
2. Check the rule precedence
   When there is a word consisting of a prefix and a suffix, including: “be-lah”, “be-an”, “me-i”, “di-i”, “pe-i”, or “te-i”, a change in the order of stemming stages is necessary, as the process should start from the prefix. Thus, the stemming stages should be in the following order (5, 6, 3, 4, 7, 8). If not so, then the order should be as follows (3, 4, 5, 6, 7, 8).
3. Inflection Suffixes
   This includes the set of sufixes that does not alter the root word: “-lah”, “-kah”, “-ku”, “-mu”, or “-nya”. If they include the particles of “-lah”,‟-kah”,”-tah” or “-pun” then this stage should be re-conducted to remove Possessive Pronouns, including the following: “-ku”, “-mu”, or “-nya”. For example, “rumahmulah” needs the removal of “-lah” to become “rumahmu”. Then, the stemming process continues to remove the Possesive Pronouns to become “rumah”.
4. Derivation Suffixes
   This includes the set of sufixes that is directly applied to root words. The suffixes responsible for the alteration in word classes include: “-i”, “-an”, or “-kan”. For example, the word “tendang”, which is a verb, shall become a noun when it receives the suffix “-an” as in “tendangan”
5. Derivation Preffix
   This includes the set of prefixes that is applied either directly to root words, or to words consisting of up to two other derivation prefixes. Derivation Prefix is the fixes put in front of a word, including “di-”,“ke-”,“se-”,“me-”,“be-”,“pe”, “te-”.
   While the suffixes are removed at stage 4, the process may continue to stage 5a, if not 5b.

   a. Unacceptable combination of prefix and suffix
      If a word consists of a prefix and a suffix as shown in Table 1, then the stage shall stop.

Table 1. Unacceptable combination of prefix and suffix

Prefix	Unacceptable Suffix
be-	-i
di-	-an
ke-	-i, -kan
me-	-an
se-	-i, -kan

   b. Identifying prefix types:
      Standard: prefixes “di-”, “ke-”, “se-”, can be removed directly.
      Complex: prefixes “me-”, “be-”, “pe”, “te-”, shall experience an alteration following the root words. This, it is necessary to consider the alteration rule before removing a prefix.

6. Recording
   If the root words are not found yet even after stage 5 is completed, then recording is due. Recording is the stage where the words received an addition of character in the beginning of the cut word, referring to the prefix alteration rule.
   For example, the prefix in “menangkap” should be altered; prefix “me-” should be removed to result in the word “nangkap”, although the word is not available in the dictionary. It fulfills one alteration rule as the character “n” is changed into “t”, altering the word into “tangkap”.

7. Restoring Confix
   Additional function of Enhanced Confix Stripping serves to resolve the problem of unnecessary suffix cutting. This process restore the word to its pre-recoding form and return all the prefixes and suffixes. This process includes:
   a. Restoring all previously removed prefixes to result in the following model : [DP+[DP+[DP]]] + Root Words. Cutting of prefixes includes searching for the words in the dictionary, followed by processing the words put in the model.
   b. Restoring confixes according to the model order in Bahasa. This means that the restoring starts from DS “-i”, “- kan”, “-an”, lalu PP “-ku”, “-mu”, “-nya”, and P “- lah”, “-kah”, “-tah”, “-pun”. For “-kan”, the restoring should specifically start from “k”, then “an”.
   c. Checking the root words into dictionary. When a word is found, it should be recorded as a root word candidate.

8. Checking Word into Dictionary : Same as step 1, but if no root word found from stemming process then restore into original word.
9. Choosing Candidate List : In this prosess is choosing final root word based on candidate list. If two candidates are found, the first root word is chosen, if more three candidates are found, the longest characters of root word is chosen. This approach is based on the result of related research [11].

2. 3 Testing

The next step after the system has been completely implemented is the testing. The testing is needed to measure the system performance. Parameter measured in the testing is the accuracy level of the algorithm in determining the correct root words. This measurement uses the following formula 1:

$$Accuracy = \frac{Correct\ root\ words\ resulted\ from\ stemming}{Number\ of\ words\ processed}\ x\ 100\% \tag{1}$$

After obtaining the test outcome data, the comparison between modified algorithm and un-modified algorithm for stemming is conducted. One-tailed McNemar method test is used to compare the accuracy between the two methods to prove that system A is better than system B statistically. One of the uses of the McNemar Test is to evaluate pre-test and post-test research designs in one group. Values for each pre-test and post-test were measured to produce two variables, true or false. The hypothesis formulation is "is there a significant difference between pre-test and post-test scores?" [12].

In this research, the result data from stemming is binary; correct for the root words found from stemming and incorrect for the words not found through stemming. The data is then calculated and grouped into four categories: correct words resulted from unmodified method, incorrect words resulted from unmodified method, correct words resulted from modified method, incorrect words resulted from modified method, as presented in Table 2.

Table 2. McNemar Test

		Modified Method		
		Correct words	Incorrect words	Total
Unmodified Method	Correct words	a	b	$a+b = n_1$
	Incorrect words	c	d	$c+d = n_2$
	Total	$a+c$	$a+d$	$n = n_1 + n_2$

In McNemar test, the observed cells include cell b and c, in which the responses to the two conditions are different.

1. Null Hypothesis = $H_0 : \pi b = \pi c$

   The proportion of observation in cell b is equal to that of cell c, so there is no difference in the results from unmodified method and modified method.
2. Alternative Hypothesis = $H_1 : \pi b \neq \pi c$

   The proportion of observation in cell b is unequal to that of cell c, so there is a difference in the results from unmodified method and modified method. This is a two-tailed hypothesis, for πc ymb value is greater or smaller than πb.

   Or

   Alternative Hypothesis = $H_1 : \pi b < \pi c$ or $H_1 : \pi b > \pi c$

   The proportion of observation in cell b is greater or smaller than that of cell c, so there is a difference in the results from unmodified method and modified method. This is a one-tailed hypothesis.

Based on the Table, the calculation uses the following formula 2:

$$x^2 = \frac{(|b - c\,| - 1)^2}{b + c} \qquad (2)$$

Result of x^2 calculation is compared to the Table of Chi-Square Distribution with degree of freedom = 1 in α = 0,05. If x^2 calculation is equal to the value in the Table of Chi-Square Distribution then H_0 is accepted**,** if there is a difference in x^2 value from the one in the Table of Chi-Square Distribution, then H_1 is accepted.

3. RESULTS AND DISCUSSION

This research conducts stemming on 15,934 words. The result is shown in Table 3. Table 2 is then used as the basis to calculate accuracy level using formula 1.

Table 3. Number of words successfully resulted from stemming

Parameter	ECS	NDT
Successfully Stemming	14.698	14.843
Failed to Stemming	1.236	1.091
Total Data	15.934	15.934
Accuration	92.2 %	93.15 %

The result of the comparison between the two of them shows that the use of non-deteministic method on the Enhanced Confix Stripping Stemmer can improve accuracy. Additionally, McNemar test used to test whether there are significant differences between the modified and unmodified stemming methods shows that the application of non-deterministic method in Enhanced Confix Striping has resulted in a significant change in accuracy.

This study examines 15,934 words. The total number of candidate words that are successfully identified through non-deterministic application is presented in Table 4.

Table 4. Total data based on the number of candidates.

Candidate Found	Total
1	14.892
2	991
3	50
4	1

Some examples of words with identified root words in the candidate list are presented in Table 5.

Table 5. Root words identified in the candidate list

no	dataset	ecs	ndt	candidate
1	teriakan.	ria	teriak	ria, teriak, riak
2	kutukan	kutu	kutuk	kutu, kutuk, tuk
3	kepada	kepada	kepada	kepada, pada
4	kutukan-Ku	kutu	kutuk	kutu, kutuk, tuk
5	kekuningan	kuning	kuningan	kuning, ning, kuningan
6	penyelidikan	lidi	sidi	sidi, selidik
7	penyelidikan,	lidi	sidi	sidi, selidik
8	penyelidikanku	lidi	sidi	sidi, selidik
9	berantakan	beranta	berantak	beranta, anta, berantak
10	teriakannya	ria	teriak	ria, teriak, riak

Table 4 shows that there are 1,042 data with more than one candidate in which they are filtered to find out the incorrect word in choosing the root word. The calculation to determine the percentage of errors in determining the root words based on the list of candidates is conducted. Table 6 suggested that the slightest error in choosing root word candidates occurs if there are 4 root word candidates found, and the greatest error occurs if there are 3 root word candidates found. Errors in the process of choosing root words in the candidate list can be investigated in further research

Table 6. Error rate in choosing root word candidates

Number of Candidates	**Total Data**	**Incorrect Data**		**Error Percentage**
		Errors in Choosing Root Word	**No candidates found**	
2	991	3	0	0.1 %
3	50	1	0	2 %
4	1	0	0	0 %

Although the implementation of the deterministic method is able to solve several problems in the Enhanced Confix Striping, the processing time will be longer. In the Enhanced Confix Striping, it takes 1,400 seconds to process 15,934 words, while it only takes 4,068 seconds using the Enhanced Confix Striping, with the addition of a non-deterministic approach. An increase in processing time on this modification is due to the non-deterministic method contained in Enhanced Confix Striping having rules that will continue to run even after the words processed have been found in the dictionary.

4. CONCLUSIONS

Based on the research and test results, it can be concluded that the modification of Enhanced Confix Striping Stemmer using non determinsitic method was able to identify the possibilities of root words that can be formed in a single word through the candidate list. This modification also can increase the accuration of Enhanced Confix Striping Stemmer.

5. FUTURE WORKS

Suggestions that can be used for further research, especially in research related to the stemming algoritm is the need to enrich the dictionary of words and add another preprocessing to handle plural's word variation. In addition, it is also necessary to think about the process of selecting root word from candidate list that matches the context of the sentence.

REFERENCES

[1] D. Suhartono,"Lemmatization technique in bahasa: Indonesian," *Journal of Software*, Volume 9 No. 5, p.1203 Jakarta, 2014 [Online]. Available : https://www.researchgate.net/profile/Derwin_Suhartono/publication/273076749_Lemmatization_Technique_in_Bahasa_Indonesian_Language/links/58e866520f7e9b978f7f550e/Lemmatization-Technique-in-Bahasa-Indonesian-Language.pdf [Accessed : 25 August 2019]

[2] D. Wahyudi, T. Susyanto, and D. Nugroho, “Implementasi dan Analisis Algoritma Stemming Nazief & Adriani dan Porter pada Dokumen Berbahasa Indonesia,” *Jurnal Ilmiah Sinus*, vol. 15, no. 2, pp. 49–56, Surakarta, 2017.

[3] A.F Hidayatullah, "The Influence of Stemming on Indonesian Tweet Sentiment Analysis," Proceeding of International Conference on Electrical Engineering, Computer Science and Informatics, Palembang, 2015 [Online]. Available : http://journal.portalgaruda.org/index.php/EECSI/article/view/791/736 [Accessed : 20 August 2019]

[4] S. S. Manase, “Studi Perbandingan Algoritma - Algoritma Stemming Untuk Dokumen Teks Bahasa Indonesia,”. *Jurnal INKOFAR*. Volume 1 No. 1, July 2017. Politeknik META. Bekasi, 2017 [Online]. Available : http://www.politeknikmeta.ac.id/meta/ojs/index.php/inkofar/article/view/2 [Accessed : 15 August 2019]

[5] S. Prasetyo, “Komparasi Algoritme Stemming Nazief & Adriani Dengan Tala Pada Teks Bahasa Indonesia,” *Tesis*. Magister Teknik Informatika STMIK Amikom. Yogyakarta, 2016

[6] D. Novitasari, "Perbandingan Algoritma Stemming Porter Denganarifin Setiono untuk Menentukan Tingkat Ketepatan Kata Dasar," Jurnal String Vol.1 No.2, Jakarta, 2016

[7] R.K. Hapsari, Y.J. Santoso, “Stemming Artikel Berbahasa Indonesia dengan Pendekatan Confix-Stripping,” *Prosiding Seminar Nasional Manajemen Teknologi XXII* , 2015 [Online]. Available : http://mmt.its.ac.id/download/SEMNAS/SEMNAS%20XXII/MTI/25.%20Prosiding%20Rinci%20Kembang%20Hapsari%20-%20Ok.pdf [Accessed : 20 August 2019]

[8] R. Setiawan, A. Kurniawan, W. Budiharto, I. H. Kartowisastro and H. Prabowo, "Flexible affix classification for stemming Indonesian Language," 2016 13th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), Chiang Mai, 2016 [Online]. Available : https://ieeexplore.ieee.org/abstract/document/7561257. [Accessed : 20 August 2019]

[9] P. Prihatini, “Stemming Algorithm for Indonesian Digital News Text Processing,”.*International Journal of Engineering and Emerging Technology*. Bali, 2017 [Online]. Available : https://ojs.unud.ac.id/index.php/ijeet/article/view/36342. [Accessed : 15 August 2019]

[10] A. Purwarianti, “A Non Deterministic Indonesian Stemmer,”. *Proceedings of the 2011 International Conference on Electrical Engineering*. Bandung, 2011 [Online]. Available : https://ieeexplore.ieee.org/document/6021829. [Accessed : 15 August 2019]

[11] W. Hidayat, “Ekstraksi Kata Dasar Secara Berjenjang (Incremental Stemming) Berbasis Aturan Morfologi untuk Teks Berbahasa Indonesia,” *Jurnal Infotel* Vol 9 No 2. Purwokerto, 2017 [Online]. Available : http://ejournal.st3telkom.ac.id/index.php/infotel/article/view/216. [Accessed : 15 August 2019]

[12] A. Heryana, “Uji McNemar dan Uji Peringkat Bertanda Wilcoxon data berpasangan”. *Materi Kuliah*, Universitas Esa Unggul. Jakarta. 2017 [Online], Available : https://docplayer.info/47771598-Uji-mcnemar-dan-uji-peringkat-bertanda-wilcoxon-data-berpasangan-ade-heryana-sst-mkm.html [Accessed : 20 October 2019]