Comparison of Nazief-Adriani and Paice-Husk algorithm for Indonesian text stemming process

J Jumadi1,*, D S Maylawati1,2, L D Pratiwi1 and M A Ramdhani1

1 Department of Informatics, UIN Sunan Gunung Djati Bandung, Indonesia
2 Faculty of Information and Communication Technology, Universiti Teknikal Malaysia Melaka, Malaysia

*jumadi@uinsgd.ac.id

Abstract. Stemming is a process contained in the pre-processing stage that recognizes basic words (stem word) by combining or solving each of the variants of a word. Every language is unique, the most popular stemming algorithm for Indonesian text is Nazief-Adriani algorithm. Therefore, this study aims to compare Nazief-Adriani algorithm with another stemming algorithm for Indonesian text, that is Paice-Husk stemming algorithm which is commonly used for English. Beside, Nazief-Adriani and Paice-Husk algorithm for stemming process, this study use McCabe Cyclometric Complexity Metrix to evaluate the complexity of algorithm. Based on the experiment result with 20 sentences as data with a thousand words, the accuracy of the Nazief-Adriani algorithm is better than the Paice-Husk algorithm, which is 91.87% compared to 64.43%. Likewise, in terms of complexity, the algorithm is still more complex Paice-Husk than Nazief-Adriani. However, in terms of processing time, the Paice-Husk algorithm is slightly faster than the Nazief-Adriani algorithm. These results indicate that the Paice-Husk algorithm requires a more complete implementation of Indonesian morphological and grammatical rules to produce the better Indonesian stem words.

1. Introduction

Today, the volume of text data that flowing every day, even every second, are very large and massive along with its development of social media, news portal, e-mail, and so on. There is a text mining technology which used to find the important information (insight knowledge) from the large text data [1–3]. Text mining has several process, begin from data collection, text pre-processing, feature extraction, feature selection, mining process, until evaluation and finding the insight knowledge [4,5]. Text pre-processing stage is one of important process, because well preparation of text data will produce a good quality result [6,7]. Stemming process is one of text pre-processing which returns the word to its basic form (stem word) [8]. Most of the words that derived by affixes (prefix, suffix, of infix) are contain the meaning of text. So that, to maintain the meaning of text, stemming process is important to be conducted. Basically, stemming algorithm begin from Porter algorithm for English [9], then developed such as Lovins algorithm [10], Paice-Husk algorithm [11], and so on. Uniquely, every language has own characteristic and structure, specially affixes structure, so stemming algorithm will be transformed in accordance with the characteristic of language.

For Indonesian language there is Nazief-Adriani algorithm that widely used and popular for Indonesian stemming [12,13]. Besides Nazief-Adriani algorithm, there are Arifin-Setiono algorithm [14], Vega algorithm [15], Ahmad-Yusoff-Sembok [16] and Bakar-Rahman [17] that build stemming
algorithm for Malaysian that quite similar with Indonesian, Asian-William-Tahaghoghi algorithm [18], Setiawan-Kurniawan-Budiharto algorithm [19], Ni’mah-Suryaningrum-Arifin [20], and so on. Every algorithm certainty has advantages and disadvantages, there are several previous related works that compare and evaluate the effect of stemming algorithm for Indonesian language, among others: (1) evaluate the stemming algorithm for information retrieval with Indonesian language [21]; (2) comparison between Nazief-Adriani and Porter stemming algorithm for Indonesian language [22]; (3) Porter stemming algorithm improvement for Indonesian language and comparing with Nazief-Adriani and Lucene algorithm [23]; (4) comparing several stemming (Nazief, Arifin, Fadillah, Asian, Enhanced Confix Stripping, Arifiyanti and Porter) and the effect of stemming process for Indonesian text [24]; (5) evaluate the stemming influence for Indonesian text from Twitter social media [25]; (6) improving stemming algorithm for Indonesian text by rearrange stemming process steps sequence [26].

Based on several previous researches above, the most Indonesian stemming algorithm comparison is conducted using Nazief-Adriani and Porter. Not many research’s that use Paice-Husk algorithm for Indonesian stemming. Therefore, in this research propose Paice-Husk stemming algorithm that basically for English, is used for Indonesian language. Furthermore, this research aims to evaluate the accuracy of Paice-Husk which compared with Nazief-Adriani algorithm.

2. Methodology
This section explains about Nazief-Adriani and Paice-Husk stemming algorithm which used for Indonesian text in this research.

2.1. Nazief-Adriani algorithm
Nazief-Adriani algorithm is a popular and widely used for stemming process of Indonesian language [18,21,22,27]. Moreover, today in Python library, there is Sastrawi library that process Indonesian text with Nazief-Adriani algorithm in the stemming function [28]. Nazief-Adriani algorithm conducts confix-stripping, encountered a complex prefix, and also encountered ambiguity [12,13]. But, comparing with Porter algorithm, Nazief-Adriani algorithm need more long processing time, although the accuracy is better [22]. Figure 1 is described the Nazief-Adriani algorithm process. Nazief-Adriani presents the confix-stripping approach with the affix order: [[[DP+]DP+]DP+] root-word [[+DS][+PP][+P]]. Where, square brackets is optional affix, DP is Derivational Prefixes, DS is Derivational Suffixes, PP is Possessive Pronouns, and P is Particle.
2.2. Paice-Husk algorithm
The Paice-Husk stemmer was developed by Chris Paice at Lancaster University. In 1980, Chris Paice died. Paice-Husk stemmer was first published in 1990 [11]. The research on stemmer was developed by Paice’s assistant named Gareth Husk. Paice-Husk stemmer states that each word is a combination of several basic words. Although Paice-Husk stemmer is known to be very efficient and is known to be very strong and aggressive. Paice-Husk stemmer uses a table or list of rules, each of which has a duty to cut additions. The cutting technique used is to avoid spelling problems unless described earlier. By replacing the suffix not by cutting the suffix, the stemmer does it without separating the steps in the stemming process. This helps maintain the efficiency of the algorithm will make it more effective.

Figure 1. Nazief-Adriani algorithm.

Figure 2. Paice-Husk algorithm [29].
3. Results and discussion
This section provides about the characteristics of Indonesian affixes, the scenario of experiment of Nazief-Adriani and Paice-Husk stemming algorithm, result of experiment and evaluation. The experiment of research only use for text pre-processing stage (stemming process) not until mining process, such as classification or clustering.

3.1. Indonesian affixes rules
There are several affixes rules or characteristics of Indonesian language, among others [13,21,23):

- Indonesian language has a prefix, suffix, infix, and confix as affixes.
- The basic prefix among others: “meng-”, “ber-”, “ter-”, “di-”, “ke-”, “pemg-”, “se-”, and “per”. Then, several of basic prefix had its allomorph, such as “me- + ‘l’ /‘m’ /‘n’ /‘r’ /‘y’ /‘w’ /‘ny’ /‘ng’”; “men- + ‘d’ /‘t’ /‘c’ /‘j’ /‘sy’”; “mem + ‘b’ /‘p’ /‘f’”; “meny- + ‘s’”; “pe- + ‘t’”; “pel-”; “be- + ‘r’”; “bel-”; “te- + ‘r’”; “pe- + ‘r’ /‘y’ /‘w’ /‘ny’ /‘ng’ /‘c’ /‘j’ /‘sy’”; “pen- + ‘d’ /‘t’”; “pem- + ‘b’ /‘p’ /‘f’”; “peny- + ‘s’”; “men- + ‘t’”; “pen- + ‘t’”; “meng- + ‘k’”; “peng- + ‘k’”; “mem- + ‘p’”; and “pem- + ‘p’”.

- There were three suffixes, among others “-i”, “-kan”, and “-an”. Besides, there were two types of inflectional suffixes, among others particles and possessive pronouns. Particles among others: “-lah”, “-tah”, and “-pun”. While, possessive pronouns among others: “-ku”, “-mu”, and “-nya”.
- Infix was inserted into the stem word, such as “-el-”, “-er-”, “-em-”, and “-er-”.
- For confix that combine prefix and suffix, there were several prefixes and suffixes that might not be combined, such as “meng- + -an”, “per- + -an”, “ber- + -i”, “ter- + -an”, “di- + -an”, and “ke- + -kan”. And also, there were combination prefixes and suffix rules, among others: prefix “meng-” or “di- + per-“ with suffix “-kan” or “-i”; and prefix “meng-“ or “di-“ or “ter- + ber-“ with suffix “-kan”.
- Last, Indonesian language can reduplicate affix form which was repetition words with a new meaning. For example “kekanak-kanakan” that mean “childish” was formed from “anak” that mean “child”.

3.2. Experiment scenario of Nazief-Adriani and Paice-Husk algorithm
Experiment is conducted with build the simple web-based application that has a function to read document or sentence, then run the Nazief-Adriani and Paice-Husk stemming algorithm. The application produces the result of stemming process. Blackbox testing is used to test and make sure the functionalities of application have been run correctly [30,31]. Besides, to evaluate the accuracy and processing time between Nazief-Adriani and Paice-Husk algorithm, there are 200 sentences in Indonesian language that prepared with 1958 words with affix. The accuracy is calculated using simple formula (1), while processing time uses seconds as a time metrics.

\[
\text{Accuracy} = \frac{\text{number of correct stemmed words matches}}{\text{number of correct and incorrect stemmed words matches}} \times 100\%
\]  

(1)

3.3. Result and evaluation of Nazief-Adriani and Paice-Husk algorithm
Table 1 provides the example of experiment result of Nazief-Adriani and Paice-Husk to evaluate the accuracy of those algorithm. The sentences or documents that processed through a series of text pre-processing phase, such as tokenizing, lowercase, remove regular expression, and stop-words removing [7,8,32,33]. It is important to minimize the size of feature and remove unused features.
Table 1. The example of Nazief-Adriani and Paice-Husk result.

| Text Data | After Pre-processing | Nazief-Adriani Result | Correct | Incorrect | Paice-Husk Result | Correct | Incorrect |
|-----------|----------------------|-----------------------|---------|-----------|--------------------|---------|-----------|
| Pengembangan bergerak dari konsep yaitu melalui desain, implementasi, pengujian, instalasi, penyelesaian masalah, dan berakhir di operasi dan pemeliharaan | pengembangan bergerak desain implementasi pengujian instalasi penyelesaian masalah dan berakhir di operasi dan pemeliharaan | kembang konsep desain implementasi uji instalasi selesai akhir operasi pemeliharaan | 11 0 | ngembang gerak konsep desain implementasi uji instalasi selesai akhir operasi pemeliharaan | 4 7 | |
| Relasi menunjukkan adanya hubungan diantara sejumlah entitas yang berasal dari himpunan entitas yang berbeda | relasi menunjukkan hubungan diantara entitas berasal himpunan entitas berbeda | relasi nunjuk hubungan antara entitas asal himpunan beda | 8 1 | relasi nunjuk hubungan antara entitas asal himpunan | 8 1 | |

Note: word with highlight means incorrect

From 200 sentences with 1958 words with affix, Nazief-Adriani produces 1799 correct stemmed words and 159 incorrect stemmed words, while Paice-Husk algorithm produces 1261 correct stemmed words and 697 incorrect stemmed words. It means that Nazief-Adriani has accuracy around 91.87%, while Paice-Husk has accuracy around 64.4%. This result shows that Nazief-Adriani stemming algorithm is better than Paice-Husk algorithm in producing Indonesian stemmed word. Basically, Paice-Husk is used for English, although in this research conduct modification to adjust to the needs and affix rules in Indonesian. However, Nazief-Adriani algorithm also produces incorrect result, especially for the affix that for affixes that melt when meeting the letter "k", “p”, “t", and “s”. Then, for processing time, the difference is not significant, Paice-Husk has average processing time around 7.36 seconds, while Nazief-Adriani has average processing time around 7.95 seconds. It is because Nazief-Adriani check the stemmed word database repeatedly. This result shows that Nazief-Adriani still better stemming algorithm for Indonesian language comparing with Paice-Husk algorithm.

4. Conclusion

Many stemming algorithms that can be used to prepare the quality of text data well before conducting the mining process. However, every stemming algorithm must be prepared in accordance with the characteristics or rules of language, including Indonesian language. One of popular stemming algorithm for Indonesian language is Nazief-Adriani. This research found that another stemming algorithm such as Paice-Husk also can be used for Indonesian language with several modifications in accordance with Indonesian affix rules. The experiment shows that Nazief-Adriani still better that Paice-Husk in producing the correct stemmed words, although the processing time is longer that Paice-Husk.
Therefore, for the further works, Paice-Husk can be enhanced to produce Indonesian stemmed word well.

Acknowledgments
We would like to thank to the Research and Publishing Center, the Research and Community Service Institute, UIN Sunan Gunung Djati Bandung for supporting and funding the publication of this research.

References
[1] Torre C J, Martín-Bautista M J S and Blanco I D 2008 Text Knowledge Mining: And Approach to Text Mining ESTYLFO8 17 19
[2] Weiss S M, Indurkhya N and Zhang T 2010 Information retrieval and text mining Fundamentals of Predictive Text Mining (London: Springer) pp 75-90
[3] Han J, Pei J and Kamber M 2011 Data mining: concepts and techniques (Amsterdam: Elsevier)
[4] Witten I H 2004 Text mining Pract. Handb. Internet Comput. pp 1-23
[5] Zohar Y E 2002 Introduction to text mining (USA: University of Illinois)
[6] Maylawati D S A, Aulawi H and Ramdhani M A 2019 Flexibility of Indonesian text pre-processing library Indonesian Journal of Electrical Engineering and Computer Science 13(1) 420-426
[7] Kannan S and Gurusamy V 2014 Preprocessing techniques for text mining International Journal of Computer Science & Communication Networks 5(1) 7-16
[8] Vijayarani S, Ilamathi M J and Nithya M 2015 Preprocessing techniques for text mining-an overview International Journal of Computer Science & Communication Networks 5(1) 7-16
[9] Porter M F 1980 An algorithm for suffix stripping Program 14(3) 130-137
[10] Lovins J B 1968 Development of a stemming algorithm Mech. Transl. Comput. Linguistics 11(1-2) 22-31
[11] Paice C D 1990 Another stemmer ACM Sigir Forum 24(3) 56-61
[12] Nazief B and Adriani M 1996 Confix Stripping: Approach to Stemming Algorithm for Bahasa Indonesia Internal publication, Faculty of Computer Science, University of Indonesia, Depok, Jakarta 41
[13] Adriani M, Asian J, Nazief B, Tahaghoghi S M and Williams H E 2007 Stemming Indonesian: A confix-stripping approach ACM Transactions on Asian Language Information Processing (TALIP) 6(4) 1-33
[14] Arifin A Z and Setiono A N 2002 Classification of event news documents in Indonesian language using single pass clustering algorithm Proceedings of the Seminar on Intelligent Technology and its Applications (SITIA) ‘ (Surabaya, Indonesia: Teknik Elektro, Sepuluh Nopember Institute of Technology)
[15] Vega V B 2001 Information retrieval for the Indonesian language (Master's thesis, Singapore: National University of Singapore)
[16] Ahmad F, Yusoff M and Sembok T M 1996 Experiments with a stemming algorithm for Malay words Journal of the American Society for Information Science 47(12) 909-918
[17] Bakar Z A and Rahman N A 2003 Evaluating the effectiveness of thesaurus and stemming methods in retrieving Malay translated Al-Quran documents International Conference on Asian Digital Libraries (Berlin, Heidelberg: Springer) pp 653-662
[18] Asian J, Williams H E and Tahaghoghi S M 2005 Stemming indonesian Proceeding of the Twenty-eighth Australasian conference on Computer Science-Volume 38 pp 307-314
[19] Setiawan R, Kurniawan A, Budiharto W, Kartowisasastro I H and Prabowo H 2016 Flexible affix classification for stemming Indonesian Language. In 2016 13th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON) (IEEE) pp 1-6
[20] Ni'mah A T, Suryaningrum D A and Arifin A Z 2019 Autonomy Stemmer Algorithm for Legal and Illegal Affix Detection use Finite-State Automata Method EPI International Journal of
Engineering 2(1) 46-55

[21] Tala F 2003 A study of stemming effects on information retrieval in Bahasa Indonesia (Netherlands: Universiteit van Amsterdam)

[22] Agusta L 2009 Perbandingan algoritma stemming Porter dengan algoritma Nazief & Adriani untuk stemming dokumen teks bahasa indonesia Konferensi Nasional Sistem dan Informatika 2009 pp 196-201

[23] Maylawati D S A, Zulfiqar W B, Slamet C, Ramdhani M A and Gerhana Y A 2018 An improved of stemming algorithm for mining indonesian text with slang on social media 2018 6th International Conference on Cyber and IT Service Management (CITSM) (IEEE) pp 1-6

[24] Rizki A S, Tjahyanto A and Trialih R 2019 Comparison of stemming algorithms on Indonesian text processing Telkomnika 17(1)

[25] Hidayatullah A F, Ratnasari C I and Wisnugroho S 2016 Analysis of Stemming Influence on Indonesian Tweet Classification Telkomnika 14(2) 665

[26] Widayanto H and Huda A 2017 Indonesian Language Stemmer Algorithm Improvement By Rearrange Stemming Process Steps Sequence eProceedings of Engineering 4(3)

[27] Jivani A G 2011 A comparative study of stemming algorithms Int. J. Comp. Tech. Appl 2(6) 1930-1938

[28] Robbani H A 2016 Sastrawi MIT

[29] Gurusamy V and Kannan S 2017 Performance analysis: Stemming algorithm for the english language International Journal for Scientific Research and Development 5 2321-613

[30] Utomo D W, Kurniawan D and Astuti Y P 2018 Teknik pengujian perangkat lunak dalam evaluasi sistem layanan mandiri pemantauan haji pada kementerian agama provinsi jawa tengah Simetris: Jurnal Teknik Mesin, Elektro dan Ilmu Komputer 9(2) 731-746

[31] Pressman R S 2011 Software Engineering: A Practitioner’s Approach 7th ed. (New York: McGraw-Hill)

[32] Harjanta A T J 2015 Preprocessing Text untuk Meminimalisir Kata yang Tidak Berarti dalam Proses Text Mining Jurnal Informatika Upgris 1(1 Juni)

[33] Pramudita H R 2014 Penerapan Algoritma Stemming Nazief & Adriani dan Similarity pada Penerimaan Judul Thesis Data Manajemen dan Teknologi Informasi (DASI) 15(4) 15