JOIN (Jurnal Online Informatika)
Volume 5 No.1 | June 2020: 81-88
DOI: 10.15575/join.v5i1.558

p-ISSN: 2528-1682
e-ISSN: 2527-9165

Optimization of Sentiment Analysis for Indonesian Presidential Election using Naïve Bayes and Particle Swarm Optimization

Nur Hayatin[1*], Gita Indah Marthasari[2], Lia Nuraini[3]
[1,2,3] Department of Informatics University of Muhammadiyah Malang, Indonesia

Article Info

Article history:

Received April 28, 2020
Revised July 03, 2020
Accepted July 14, 2020
Published July 15, 2020

Keywords:

Naïve Bayes
Opinion extraction
Sentiment analysis
Particle swarm optimization

ABSTRACT

Twitter can be used to analyze sentiment to get public opinion about public figures to find a trend in positive or negative responses, especially to analyze sentiments related to presidential candidates in the 2019 election in Indonesia. Naïve Bayes (NB) can be used to classify tweet feed into polarity class negative or positive, but it still has low accuracy. Therefore, this study optimizes the Naïve Bayes algorithm with Particle Swarm Optimization (NB-PSO) to classify opinions from twitter feeds to get a good accuracy of public figures sentiment analysis. PSO used to select features to find optimization values to improve the accuracy of Naïve Bayes. There are four steps to optimize NB using PSO, i.e., initializing the population (swarm), calculate the accuracy value that matched with selected features, selected the best accuracy of classification, and updating position and velocity. From this study, the group of tweets was obtained based on the positive and negative sentiments from the community towards two Indonesia presidential candidates in 2019. The NB-PSO test shows the accuracy result of 90.74%. The result of accuracy increases by 4.12% of the NB algorithm. In conclusion, the inclusion of the Particle Swarm Optimization algorithm for Naïve Bayes classification algorithm gives a significant accuracy, especially for sentiment analysis cases.

Corresponding Author:

Nur Hayatin,
Department of Informatics,
University of Muhammadiyah Malang,
Jl. Raya Tlogomas 246 Malang, Indonesia
Email: noorhayatin@umm.ac.id

1. INTRODUCTION

Twitter is a social media that contains a collection of messages about opinions, expressions of emotions. The benefit of Twitter is not only as a medium of information and communication but also as a medium for expressing public opinion. In Indonesia, especially in the election moment, twitter is usually used for political activities, namely campaign, to criticize specific policies, and the teams from various parties were busy plunging into social media to increase the popularity of their candidates [1]. People in social media, especially Twitter, are free to give their opinions about public figures. This opinion is not only in the form of positive response but also negative ones. And this can be used to analyze tweet feeds to get a sentiment of public figure's opinions to find a trend in positive or negative responses.

Sentiment analysis is the process of understanding, extracting, and processing textual data automatically to get sentiment information contained in the opinion sentences. Then classify the various polarities of the text in the sentences or documents in the two possibilities class, either positive or negative.

The research on sentiment analysis, which is to obtain opinions on public figures, had been carried out based on previous research. Research [1] uses the Lexicon-Based Approach (LBA) to sentiment analysis of tweets in the election result. But the results of the study indicate that the value of the f-measure evaluation is still low by 55%. Classification is applied to perform a sentiment analysis of twitter data [2]. Naïve Bayes (NB) is a very popular algorithm for document classification and detecting sentiment from text. The basic idea is to combine words and categories probability in estimating the possibility of the document groups. Mukherjee et al. [3] use the Naïve Bayes algorithm to detect sarcasm from 5000 tweets that represent consumer comments on several types of products. The purpose of classification is to determine whether consumer comments belong to sarcasm or not. Preprocessing is performed on all data sets, including re-tweet removal and feature extraction. Classification using Naïve Bayes gives better accuracy compared to the Maximum Entropy algorithm. Rana, et.al compared the performance of the Naïve Bayes algorithm with the Support Vector Machine (SVM) and found that classification accuracy using SVM was better [4]. The weakness of Naïve Bayes algorithm is its very sensitive to the selection of features and probability estimation results that cannot always run optimally, thus the process resulting in low accuracy.

Particle Swarm Optimization (PSO) algorithm, as a simple method in the feature selection process, can find optimization values [2][3]. Using PSO to optimize Naïve Bayes is expected to improve the accuracy to overcome the weakness of the Naïve Bayes to sentiment classification. Several studies use Particle Swarm Optimization to improve the accuracy of the classification results from Naïve Bayes. The use of PSO and NB is to classify email content as spam and non-spam [7], hoax classification [8], and other research analyzed the customer review on online marketing companies [6] and to classified digital news content taken from the site www.kompas.com (one of the largest online news in Indonesia) to some categories like gossip, culinary, and travel categories [5]. The application of Particle Swarm Optimization (PSO) is proven to increase accuracy in the classification of public opinion reviews of artist news to identify between positive reviews and negative reviews [9].

In this study, we were optimizing Naïve Bayes by applying the feature selection method using Particle Swarm Optimization to improve the accuracy from the result analyze the tweet feeds to know public opinion from the presidential candidates in Indonesia Election moment 2019.

2. METHOD

The proposed system in Figure 1 shows there are four main stages of the process, i.e., preprocessing, feature extracting, classifier model, and testing. The Corpus was taken from twitter by using twitter API to crawling data. After that, the system processed data to find out the pattern and tested the data from the learning train.

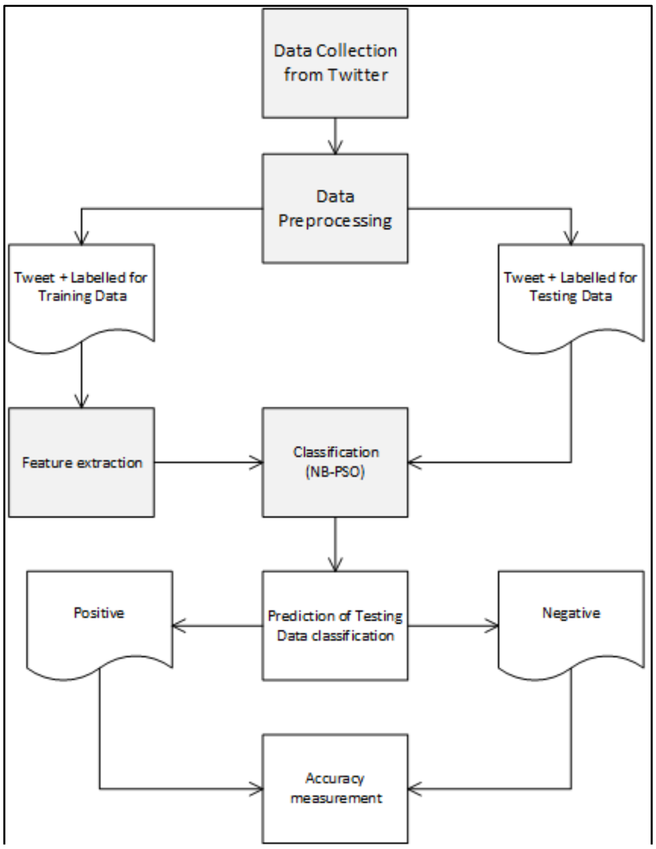

Figure 1. Proposed System

2.1. Data Collection

This stage is to collect and keep tweet data from each public figure that will be analyzed. The data in this study consisted of data from tweets obtained by crawling through web crawlers by accessing the API from Twitter. The tweet is in the Indonesia context. These tweets are searched based on keywords from the list of names of the presidential election candidate in 2019. There were four names of public figures to be analyzed, i.e., *Joko Widodo, Ma'ruf Amin, Prabowo Subianto*, and *Sandiaga Uno*. After the tweet data collection was complete, the data were divided into two groups, namely, training and testing data. There were 200 data tweets for data train and 100 data tweets for data test which derived from each public figures. In conclusion, there were 800 data tweets as the data train and 400 tweets as data test material. Therefore, there were 1200 data tweets from six public figures to analyze in this study.

Table 1 shows tweet samples from two names of public figures ("Jokowi" and "Prabowo") that is used in this research. The table shows that the tweets of those public figures contain negative and positive sentiment.

Table 1. Tweets with the keyword "Jokowi" and "Prabowo"

Tweet	Keyword	Sentiment
@D4Ni3L_Pu: Non Muslim haram masuk dan ceramah di Masjid hny berlaku buat pdkg Jokowi	Jokowi	Negative
@azelpuspita: Melihat kebersamaan dan kebahagiaan Bapak Jokowi dan Ibu Iriana membuat hati adem, kaya ada acnya	Jokowi	Positive
Gimana ini tim Prabowo-Sandi, dulu pas belum ketahuan hoaxnya dibelain mati2an setelah sudah ditangkap malah diabaikan.... #indonesiamaju	Prabowo	Negative
Orang baik mendukung Prabowo Sandi #SulselPilihPrabowoSandi	Prabowo	Positive

2.2. Data Preprocessing

Data processing consists of manual labelling and data processing to transform the data into structured data. Manual labelling is an analysis of determining positive and negative sentiments in tweets. This process covers reading intently every sentence in the tweets and adjusts sentiment tweets based on the appearance of adjectives and verbs. While preprocessing is the cleaning process and preparing data for the next process, i.e., the classification process. The reprocessing is the primary process that determines optimal results. For preprocessing, we use the text mining process that consists of case folding, tokenizing, normalization, stopwords list removal, and stemming. We use Rapidminer tools for preprocessing. Figure 2 shows the stage of text processing.

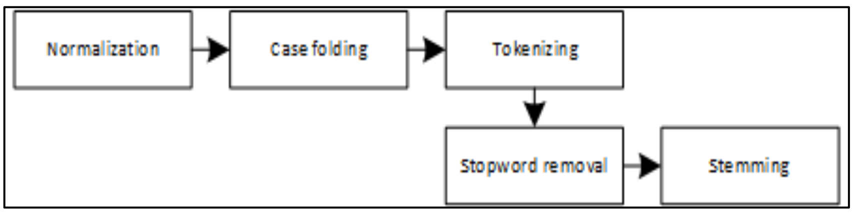

Figure 2. The stage of Text Preprocessing

The first step for preprocessing is the normalization stage, manually that includes eliminating non-important tags, mentioning that starts with the @ symbol, hashtag that starts with symbol #, and retweeting that starts with the RT, abbreviations and also slang words and regional languages which later by the author are changed to Bahasa. Case folding used to convert letter characters to lowercase letters, eliminate numeric characters, and eliminate symbol delimiter, i.e. (.), (,), (:), (;), (?), (!), (#). For this study, tokenizing used to decipher sentences into words with a space separator. The next step is eliminating the words that are considered stopwords that are made in the stopword list document. If a word is listed in the stopwords list, the word is deleted from the description. Therefore, the remaining words in the description characterize the content of a document or keywords [10]. In the stopword removal process, the term filtration process based on the stopword list serves to reduce the dimensions of terms in words that are not meaningful and have a high frequency. The last stage for preprocessing is stemming. This process converts a word into the basic form to eliminate the affixes in each word. For the steaming process, the Nazif-Adriani stemmer is used to process root words in Bahasa. The mechanism of the stemming process is based on the research [11].

2.3. Feature Extraction

The result from preprocessing is a bag of different terms called features in data classification. This bag of different terms composes both general and specific terms, which form two parts, i.e., positive and negative sentiments [12]. The specific terms in this study refer to the words that are often used in the elections moments and represent positive and negative sentiments (Table 2 shows the example of the specific term). On the other hand, general terms refer to all words from preprocessing except specific terms.

Table 2. Sentiment terms

Sentiment	Specific Term
Negative	*jelek, kalah, kecewa, buruk* (in Indonesian) bad, lose, disappointed, ugly
Positive	*dukung, idola, kagum, suka* (in Indonesia) support, idol, amazed, like

2.4. Classification using NB-PSO

For the classification process, this research used the Naïve Bayes (NB) algorithm that combined with Particle Swarm Optimization (PSO) to optimize the result. In general, the Naïve Bayes algorithm is used to determine the class of each tweet to positive and negative. Each sentence's class depends on the probability calculation result of Naïve Bayes formula [13]. A tweet would belong to the positive class if its probability value for the positive class were higher than the negative one. Conversely, a tweet of negative class occurs when its probability value for the negative class is higher than the positive one.

For this research, the Naïve Bayes algorithm will obtain classification accuracy according to feature selected based on Particle Swarm Optimization (PSO). PSO is a population research method that uses a population (swarm) of individuals (particles) that are updated from each iteration performed. In making the particle reach its optimum solution, each particle moves in the direction of the previous best position (pbest) and the best global position (*gbest*) [6]. The speed and position of the particles can be continuously updated by iterating.

There are four steps to optimize NB using PSO, i.e., initializing the population (swarm), calculate the accuracy value that matched with selected features, selected the best accuracy of classification, and updating position and velocity. Figure 3 shows the steps to optimize Naïve Bayes using PSO to get public opinion in sentiment analysis.

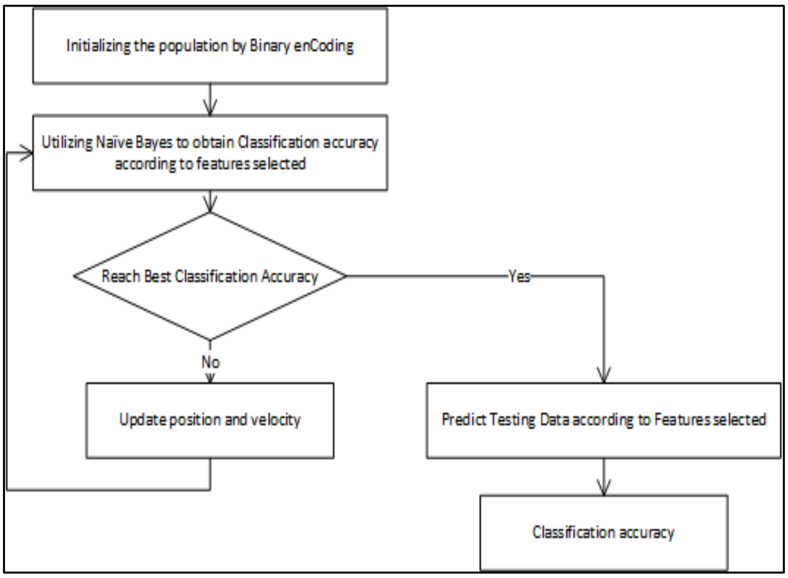

Figure 3. NB-PSO Algorithm for sentiment analysis

The first stage is initializing *X* population. Then removed the unselected attribute from the sample attribute of training data to get training data *T* according to the features of each selected particle. Calculate three probability values, i.e., the prior probability of each training data class $P(C_i)$, the conditional probability of each attribute division $P(X \mid C_i)$, and the prior probability of each class with follows $P(C_i)*P(X \mid C_i)$.

The next step is to choose the maximum prior probability $P(C_i) * P(C \mid X_i)$, which belongs to class *x*. Then calculating all classification accuracy samples as *BestAccuracy*, and selecting features according to *Bestf*. For each particle, comparing the classification accuracy of the current position with the classification accuracy of the best position *Bestpi* (feature selection) experienced. If the first is better than the last, then *Bestpi* is the same as the current position. For each particle, compare the classification accuracy of the best position *Bestpi* with the global best position *Bestpg*. If the first is better than the last, then *Bestpg* is the same as the current position. Updating the position and speed of each particle then continues to calculate $P(C_i)$. Repeat the steps until you get the *BestAccuracy* and Bestf values to get the best accuracy *BestAccuracy_temp* and a subset matched with feature selection. If *BestAccuracy_temp* > *BestAccuracy*, then *BestAccuracy* = *BestAccuracy_temp*, and *Bestf* = *f_temp*.

3. RESULTS AND DISCUSSION

This study used *Rapid Miner* tools to implement the model. Rapid Miner is a popular tool used for data analysis because it provides an integrated platform between text mining, machine learning, and predictive analytics [14], [15]. The total data is 1200 tweets from the six public figures. That is 800 for train data and 400 for test data. Performance measurement is done by comparing the accuracy results of Naive Bayes and Naïve Bayes with PSO. A gold standard of correct tweet polarity created via manual annotation to evaluate the system performance.

For measuring the performance system, each data will be tested using 10 fold cross-validation. This method has implemented to avoid data overfitting [3]. The 10 value was chosen because it is commonly used in the training data classification process. Where the 10-fold cross-validation method will divide the data into two parts, i.e., training and testing data,, where each k-fold will get the same amount of data. This process divided the data randomly into 10 parts. The testing process began with the formation of a model with the data in the first part. The form model was tested on the remaining 9 data sections. The next step was the accuracy process by calculating the amount of classified data. The 10-fold cross-validation method is determined based on the results of the researcher's experiment to get the highest accuracy results in this case what will be tested to improve the results of accuracy is the value of validation.

The Results of Naïve Bayes accuracy with the 10-fold cross-validation is presented in Table 3. The table shows that the average accuracy for the Naïve Bayes algorithm is more than 80% with the lowest accuracy value is 83%. In comparison, the highest accuracy value of Naïve Bayes is 86.62% for validation 7.

Table 3. The Results of Accuracy with the 10-fold Cross-Validation

Validation	Accuracy (%)
2	83.00%
3	84.63%
4	85.12%
5	85.12%
6	86.16%
7	**86.62%**
8	85.75%
9	85.38%
10	85.75%

For the measurement performance of Naïve Bayes with the Particle Swarm Optimization algorithm, we combine some parameters to get an optimal result. The training value in this study was determined by adjusting some parameter values of the Particle Swarm Optimization algorithm, i.e., population size (*Q*), inertia Weight (*w*), and the maximum number of generation, to obtain the highest accuracy results.

The following are the results of experiments that have been conducted to determine the value of training. The first experiment was conducted by changing the parameter value of the population size *Q* from 1 to 10 with an inertia weight of 0.1 and the maximum number of generation 30. The following Table 4 shows the results of the Naïve Bayes algorithm based on PSO accuracy for the first experiment. From the table, we can see the lowest accuracy is 86.25% for *Q*=3, and the highest accuracy is 90% for *Q*=10. So, from the result of the first experiment, we can conclude that increasing population size *Q* could not give effecting directly of accuracy value.

Table 4. Accuracy Results of Population Size Experiment

Q	Max. Number of Generation	*w*	NB-PSO Accuracy
1	30	0.1	87.50%
2	30	0.1	87.08%
3	30	0.1	86.25%
4	30	0.1	88.33%
5	30	0.1	89.17%
6	30	0.1	86.67%
7	30	0.1	89.17%
8	30	0.1	89.58%
9	30	0.1	88.75%
10	**30**	**0.1**	**90.00%**

In the second experiment, we changed the parameter value of inertia weight w in range 0.1 to 1.0, with population size 10. The population size Q that we used is 10 for the highest accuracy result based on the first experiment. Table 5 shows the accuracy result of the second experiment. The table shows the changing values of inertia weight in range 0.1 to 1.0 do not affect the accuracy result. And the accuracy for inertia weight w=0.1 to 1.0 for Q=10 and the maximum number of generation 30, it produced the same accuracy value that is 90%.

Table 5. Accuracy Results of Inertia Weight Experiment

Q	Max. Number of Generation	*w*	NB-PSO Accuracy
10	30	0.1	90.00%
10	30	0.2	90.00%
10	30	0.3	90.00%
10	30	0.4	90.00%
10	30	0.5	90.00%
10	30	0.6	90.00%
10	30	0.7	90.00%
10	30	0.8	90.00%
10	30	0.9	90.00%
10	30	1.0	90.00%

For the third experiment, we changed the parameter value of the maximum number of generations. In this experiment, we used Q=10 and inertia weight 0.1 based on the first and second experiments. While the maximum number of generations that is used in range 30 to 120. Table 6 shows the result of the third experiment. According to the table, the lowest accuracy occurs on the value with the maximum number of generation 30, which is equal to 90%. Whereas for the maximum number of generation in range 40 to 120 produces the same accuracy value of 90.83% for Q=10 and w=0.1.

Table 6. Accuracy Results of Maximum Number Generation Experiment

Q	Max. Number of Generation	w	Naïve Bayes + PSO Accuracy
10	30	0.1	90.00%
10	40	0.1	90.83%
10	50	0.1	90.83%
10	60	0.1	90.83%
10	70	0.1	90.83%
10	80	0.1	90.83%
10	90	0.1	90.83%
10	100	0.1	90.83%
10	110	0.1	90.83%
10	120	0.1	90.83%

Furthermore, the results of the testing accuracy will be represented by a matrix table using the Confusion Matrix. Using this matrix can avoid the spurious result of the classification process [3]. Based on the evaluation result, there are 564 data that are predicted to be included in the positive class, and 68 data are included in the negative class. Whereas 129 data are predicted to be negative classes according to those included in the negative class, and as many as 39 data are predicted, the negative class turns out to be included in the positive class. The results experiment for this study can be seen in table 7.

Table 7. The Comparison of Performance Measurement

Measurement	NB	NB-PSO
Accuracy	86.62%	90.83%
Precision	95.35%	97.00%
Recall	89.24%	91.58%
F-Measure	0.9134	0.9406

The algorithm proposed can increase the accuracy of the Naïve Bayes algorithm. The significant increase in measurement performance results can be shown from the graph in figure 4. The results of the accuracy with the 10-fold cross-validation have the accuracy of Naïve Bayes of 86.62%. Whereas in the accuracy, the results of the maximum number generation experiment were 90.74%. The result of accuracy increases by 4.12%. So, it can be concluded that the Particle Swarm Optimization algorithm for the Naïve Bayes classification algorithm, especially for sentiment analysis cases. Nevertheless, this study requires improvement, especially in the preprocessing section on the normalization stage due to its manual process.

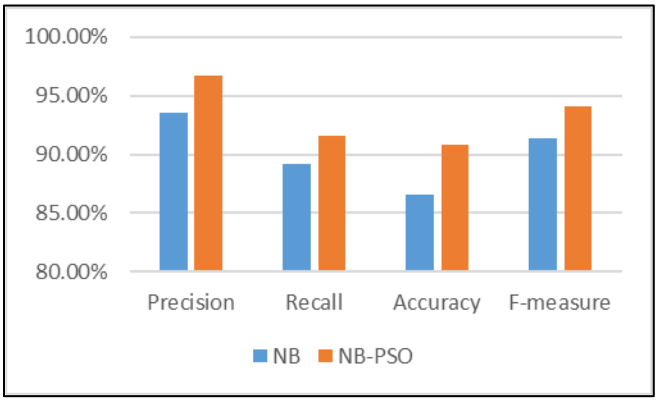

Figure 4. Comparison of Graphic of Measurement Performance

4. CONCLUSION

This study applied the feature selection method with Particle Swarm Optimization. This method improves the accuracy to overcome the weakness of the Naïve Bayes to sentiment classification. Based on the evaluation results using the confusion matrix, it is proven that the results of Naive Bayes with PSO for election sentiment analysis is 90.74%. It can be concluded that the proposed method can improve the accuracy by 4.12% from result analysis of tweet feeds to know public opinion from presidential candidates in the Indonesia Election moment 2019.

ACKNOWLEDGEMENTS

This research was supported/partially supported by the University of Muhammadiyah Malang (UMM). We thank our colleagues from the Department of Informatics UMM.

5. REFERENCES

[1] R. Rezapour, L. Wang, and J. Diesner, “Identifying the Overlap between Election Result and Candidates ’ Ranking based on Hashtag-Enhanced, Lexicon-Based Sentiment Analysis,” 2017.

[2] R. D. Desai, “Sentiment Analysis of Twitter Data,” in *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*, 2018.

[3] S. Mukherjee and P. K. Bala, “Detecting sarcasm in customer tweets: an NLP based approach.,” *Ind. Manag. Data Syst.*, vol. 117, no. 6, 2017.

[4] S. Rana and A. Singh, “Comparative Analysis of Sentiment Orientation Using SVM and Naïve Bayes Techniques,” in *2016 2nd International Conference on Next Generation Computing Technologies (NGCT)*, 2016, no. October, pp. 106–111.

[5] S. Ernawati, “Penerapan Particle Swarm Optimization untuk Seleksi Fitur pada Analisis Sentimen Review Perusahaan Penjualan Online Menggunakan Naïve Bayes,” 2016.

[6] A. Nurhadi, “Implementasi Algoritma Naïve Bayes Classifier Berbasis Particle Swarm Optimization (PSO) untuk Klasifikasi Konten Berita Digital Bahasa Indonesia,” 2016.

[7] K. A. ; T. Kumar, “Email Spam Detection Using Integrated Approach of Naïve Bayes and Particle Swarm Optimization,” in *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS).*

[8] A. P. W. ; H. A. Santoso, “Improving The Accuracy of Naïve Bayes Algorithm for Hoax Classification Using Particle Swarm Optimization,” in *2018 International Seminar on Application for Technology of Information and Communication*, 2018.

[9] A. Idrus, “Sentiment Analysis Of State Officials News On Online Media Based On Public Opinion Using Naive Bayes Classifier Algorithm And Particle Swarm Optimization,” *2018 6th Int. Conf. Cyber IT Serv. Manag.*, no. Citsm, pp. 1–7, 2018.

[10] L. H. Nguyen and A. Salopek, “A Natural Language Normalization Approach to Enhance Social Media Text Reasoning,” in *In 2017 IEEE International Conference on Big Data (Big Data)*, 2017, pp. 2019–2026.

[11] A. Rahmatulloh, N. I. Kurniati, I. Darmawan, A. Z. Asyikin, and D. W. J, “Comparison between the Stemmer Porter Effect and Nazief-Adriani on the Performance of Winnowing Algorithms for Measuring Plagiarism,” *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 9, no. August, pp. 1124–1128, 2019.

[12] C. Fiarni, “Sentiment Analysis System for Indonesia Online Retail Shop Review Using Hierarchy Naive Bayes Technique,” in *2016 4th international conference on information and communication technology (ICoICT)*, 2018, no. May 2016, pp. 1–6.

[13] J. O. Akella and L. N. Y. Akella, “Sentiment Analysis Using Naïve Bayes Algorithm,” in *2018 3rd International Conference on Inventive Computation Technologies (ICICT)*, 2018, pp. 1–4.

[14] A. Jeyaraj and M. J. Nithya, “Comparison of Feature Selection Strategies for Classification using Rapid Miner,” *Int. J. Innov. Res. Comput. Commun. Eng.*, vol. 4, no. January, pp. 13556–13563, 2018.

[15] T. Puyalnithi, M. Viswanatham, and A. Singh, “Comparison of Performance of Various Data Classification Algorithms with Ensemble Methods Using RAPIDMINER,” *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 5, no. April, pp. 1–6, 2016.