Feature Selection using Variable Length Chromosome Genetic Algorithm for Sentiment Analysis

Tirana Noor Fatyanosa[1], Fitra A. Bachtiar[2], and Mahendra Data[3]
Faculty of Computer Science
Brawijaya University
Malang, Indonesia
[1]fatyanosa@gmail.com, [2]fitra.bachtiar@ub.ac.id, [3]mahendra.data@ub.ac.id

***Abstract*— Research in sentiment analysis often have very high features for the classification and it might affect the model's accuracy. In this paper, the Variable Length Chromosome Genetic Algorithm with Naïve Bayes (VLCGA-NB) is utilized to analyze the twitter sentiment. The tweets are preprocessed in several steps before using it in the algorithm. The preprocessing conducted to reduce the number of features. After the preprocessing performed, the features that produce a higher fitness value is selected. There are five classes: Very Positive, Positive, Neutral, Negative, and Very Negative to be classified. Comparison of NB and VLCGA-NB is conducted to show the models accuracy. The experiments show that VLCGA-NB produces the higher value for the Best F-Measure results of each sentiment, lower number of features (188 features), higher accuracy (75.2%), and higher fitness value (3.088953) than NB.**

Keywords*— Twitter Sentiment Analysis, Genetic Algorithm, Naïve Bayes, feature selection

I. Introduction

These days, most people use social media to communicate with each other. Social media has grown as a well-known and useful tool to share their daily sentiments in a society. One of the social media used by most users is Twitter. Based on [1], the monthly active user of Twitter reaching a total of 300 million with 500 million daily tweets. Twitter allows the user to post tweet up to 280 characters and the ability to upload photos and short videos.

These tweets become useful data to do sentiment analysis. Numerous research regards the sentiment analysis increases. Sentiment analysis becomes one of the topics that attract the researchers with various issues, such as, politic [2], image sentiment [3], natural disaster sentiment [4], and rating review prediction [5]. However, there are several challenges behind the research related to this topic. Such as the preprocessing steps and hardly classified the tweets [6, 7].

Other problems arise in this field of study is numerous feature [8] used or generated from the sentiment. High dimension features may increase classification accuracy. However, high dimension features could decrease the classification accuracy since many irrelevant features might be used during the training process [9]. There is a need to use a good feature optimally to enhance accuracy result.

GA has been proven as a robust algorithm to select the best features for various issues. For image retrieval, [10] use GA to select the best features from 78 features in images. The result shows that GA able to improve accuracy. GA also used by [11] to select the optimal features of the network intrusion features. For mechanical fault classification, [12] uses GA to enhance the classification result by selecting the best distance-based features.

A general step performed in sentiment analysis is preprocessing. The common preprocessing steps are as follows: removing URL, removing duplicate lines, removing the unrelated word, twitter name, similar lines, punctuation, number, irrelevant tweets, hashtag, check the slang word and change it to the formal one [2]. Another preprocessing step uses the Term frequency-inverse document frequency (TF-IDF), stemming, stop words elimination, tokenization, and feature selection using WEKA's filter [6]. Preprocessing step is crucial since it could help increase the accuracy. Nonetheless, many previous studies did not consider preprocessing step thoroughly [13–15].

NB is a common algorithm that used by researchers nowadays to classify the sentiment of social media. Parveen and Pandey [16] implemented NB on movie dataset. Masrani and Poornalatha [17] use modified NB to analysis the Twitter sentiment on various topics. Goel, Gautam, and Kumar [18] use NB to classify the movie review and produce 58.4% accuracy.

Based on the information described above, Naïve Bayes (NB) is used as the classification algorithms with the unigram language model in this study. Preprocessing phase in this study involving 14 steps such as remove the irrelevant tweets, change the contractions, remove URL, number, hashtags, stop words, punctuation, non-English words, non-alphabetic characters, the words smaller than two characters, features with extremely low frequency, change all tweets to lowercase, and use lemmatization. Then, in this research, the Variable Length Chromosome GA (VLCGA) is used to determine types of sentiment in a tweet based on the features examination. The topic of this research is tweets related to self-driving cars.

II. The Proposed Variable Length Chromosome Genetic Algorithm-Naïve Bayes

A. Datasets

The Twitter data used in this research are related to self-driving cars. The data is obtained from [23] which consist of 7015 data rows. The sentiment dataset is manually classified by contributors from Figure Eight as very positive, positive, neutral, negative, very negative, and not relevant. From the dataset, we only take the sentiment and text columns in the experiments. In this research, we do not use the 'not relevant' tweets.

978-1-5386-7407-9/18/$31.00 ©2018 IEEE

TABLE I. TWEETS COMPOSITION

Class	Total	Training Data	Testing Data
Very Positive	459	430	29
Positive	1444	1369	75
Neutral	4245	3891	354
Negative	685	649	36
Very Negative	110	104	6
Total Tweets	**6943**	**6443**	**500**

We split the process after preprocessing steps into two stages: training and testing. In the training stage, we create the model. In the testing stage, the model applied and evaluated using the fitness value to know the effectiveness of the algorithm. To avoid the overfitting, we split the dataset into two datasets: training data and testing data. The training and testing data was chosen randomly. The number of data training is 6443 tweets, while the number of data testing is 500 tweets. The proportion of training data and test data is shown in Table I.

B. Preprocessing

The preprocessing steps conducted as follows:

1. *Remove the 'not relevant' tweets*: The contributors marked the tweets that are not related to the topic 'self-driving cars' with 'not relevant' class. So, in this research, we removed all tweets with 'not relevant' class. The data rows reduced from 7015 to 6943.
2. *Change the contractions to the real word or phrase*: A word or phrase that has been abbreviated by dropping at least one letters and replaced with an apostrophe is called a contraction. We changed the contractions with the real word or phrase using the list from [24].
3. *Remove URL*: All URLs was removed within tweets to reduce the number of features.
4. *Remove number*: All numbers are removed because they have no effect on sentiment analysis.
5. *Remove hashtags*: A hashtag within the tweet used to categorize topics on Twitter. In this research, hashtags are removed from tweets.
6. *Remove stop words*: In English, stop words do not contain an imperative meaning to be utilized as a part of tweets. As stop words return an extensive number of non-essential information, we removed all stop words from tweets.
7. *Remove punctuation*: In writing, punctuation is a mark that helps readers to understand the writing. Removing punctuation step is conducted because we consider that punctuation has no impact on Twitter sentiment analysis.
8. *Using lemmatization*: Lemmatization is the process of return word into its base word or lemma. The purpose of using lemmatization is to remove inflectional structures or endings [25]. We use lemmatizer from NLTK to do a full morphological investigation to precisely distinguish the lemma for each word.
9. *Remove non-English words*: We use the words corpus from NLTK to remove all non-English words. This step is conducted to reduce the number of features.
10. *Remove non-alphabetic characters*: All non-alphabetic characters were removed from this research because it will have no impact on the analysis.
11. *Change all tweets to lowercase*: The tweets are changed to lowercase to make sure there are no duplicate words because of variety in capitalization.
12. *Remove the words smaller than three characters*: We only use the words more than two characters to reduce the number of features that have no impact on sentiment analysis.
13. *Eliminating features with extremely low frequency*: The feature elimination step is conducted by counting the frequency of each feature in all training data. If the feature shows more than equals to the minimum frequency, then the feature will be saved. Otherwise, the feature will be eliminated. We compared the minimum frequency from 1 to 20 frequency (see Table II and Fig. 1).

Because this problem is searching for maximum values, the fitness value for each minimum frequency can be calculated as in (5). The higher the fitness value, the better. Features with the minimum frequency of 6 obtained the highest fitness value. Therefore, we use features with minimum frequency more than equal to 6. The number of features reduced from 4528 to 1086.

C. Variable Length Chromosome Genetic Algorithm

The conventional Genetic Algorithm (GA) usually used fixed length chromosomes based on the number of features in the dataset. However, this method has shortcomings as it needs a long computational time [19–21]. Several studies use VLCGA to reduce the number of features, such as the detection system of network intrusion [19], the optimization of topology [20], the optimization of path problems [22], and the planning of robotic path [21].

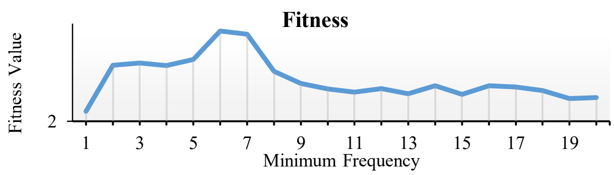

Fig. 1. Minimum Frequency Fitness Comparison

TABLE II. MINIMUM FREQUENCY COMPARISON

Minimum Frequency	Number of Word Features	Accuracy	F-Measure Very Positive	F-Measure Positive	F-Measure Neutral	F-Measure Negative	F-Measure Very Negative	Fitness
1	4528	0.676	0.052632	0.307692	0.818543	0.206897	0	2.061763
2	2587	0.692	0.097561	0.340136	0.827493	0.387097	0	2.344287
3	1878	0.714	0.04878	0.4	0.850606	0.34375	0	2.357136
4	1486	0.706	0.139535	0.362319	0.843792	0.290323	0	2.341968
5	1248	0.712	0.142857	0.353846	0.848806	0.322581	0	2.38009
6	1086	0.718	0.095238	0.353846	0.848883	0.357143	0.181818	2.554928
7	948	0.72	0.097561	0.341085	0.853018	0.357143	0.166667	2.535474
…	…	…	…	…	…	…	…	…
20	373	0.704	0.142857	0.25	0.84399	0.204082	0	2.144929

this reason, the VLCGA is utilized in this research. The chromosome length is randomly generated with genes that are filled with randomly selected features after the preprocessing steps. As a result, the initial population will consist of different chromosomes lengths. Because the implementation of union crossover, intersection crossover, and mutation process in this research, these chromosomes length will also change (increase or decrease) during the iteration.

Moreover, there is no limitation to the number of features used in the experiments. But, this research aims to select the best features with a minimum number of features that produces high accuracy, high F-Measures for each sentiment, and high fitness value.

VLCGA process consists of initialization population, fitness calculation, crossover, mutation, and selection as shown in Fig. 7. A trained model produced by NB based on the training data and features in the chromosome is used to classify the testing data. The fitness value is then calculated based on the classification result (see Fig. 8).

1. *Chromosome Representation*: In this research, we use the Variable Length Chromosome Genetic Algorithm (VLCGA) to select the best features. The chromosome consists of features with different quantity (see Fig. 2).

two	place	invest	money	could
printing	self	driving	car	

Fig. 2. Chromosome example

The genes for each chromosome was chosen randomly from 1086 selected features after the preprocessing steps. Each chromosome length for the first generation also chosen randomly from 1 to the total number of features.

2. *Crossover*: There are two types of crossover that utilized in this research such as Union Crossover and Intersection Crossover.

a) Union Crossover:

The union crossover produced a child that contains all of the features that are in at least one of the two parents. For example, the selected parents are in Fig. 3 and Fig. 4. Then, the crossover is done by copy all genes distinctly from both parents. The child can be seen in Fig. 5.

two	place	invest	money	could
printing	self	driving	car	

Fig. 3. Parent 1

saw	self	driving	car	painted	green

Fig. 4. Parent 2

two	place	invest	money	could	printing
self	driving	car	saw	painted	green

Fig. 5. Child

b) Intersection Crossover

The intersection crossover produced a child that contains all of the features that are in both parents. For example, the selected parents are in Fig. 3 and Fig. 4. Then, the crossover is done by copy all genes that contained in both parents. The resulted child can be seen in Fig. 6.

self	driving	car

Fig. 6. Child

3. *Mutation*: The mutation we use in this research is changing each gene of the selected chromosome with the new feature if the gene's random value is more than the mutation rate (MR). The number of the child produced by the mutation process determined by the multiplication between the mutation rate (MR) and the total population. For example, we define the MR by 0,3 and the total population by 100. Then, the total of the child produced by the mutation process is total population * MR = 100 * 0,3 = 30.

4. *Selection*: Based on [26], elitist selection has the ability to maintain population diversity, explore the search space and avoid the local optima. So that in this research, we use elitist selection.

5. *Fitness Function*: The fitness function we use in the experiment consists of the addition of accuracy and F-Measure of each sentiment after the NB classification as shown in Fig. 8. The Accuracy formula is shown in (1), the Precision formula is shown in (2), the Recall formula shown in (3), the F-Measure formula shown in (4), and the Fitness function shows in (5).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$F_1 = 2\,x\,\frac{Precision\ x\ Recall}{Precision+Recall} \quad (4)$$

$$Fitness = Accuracy + F_{1_{vpos}} + F_{1_{pos}} + F_{1_n} + F_{1_{neg}} + F_{1_{vneg}} \quad (5)$$

Where:

$F_1 = F - measure$
$TP = True\ Positive$
$TN = True\ Negative$
$FP = False\ Positive$
$FN = False\ Negative$
$F_{1_{vpos}} = F - measure\ for\ very\ positive$
$F_{1_{pos}} = F - measure\ for\ positive$
$F_{1_n} = F - measure\ for\ neutral$
$F_{1_{neg}} = F - measure\ for\ negative$
$F_{1_{vneg}} = F - measure\ for\ very\ negative$

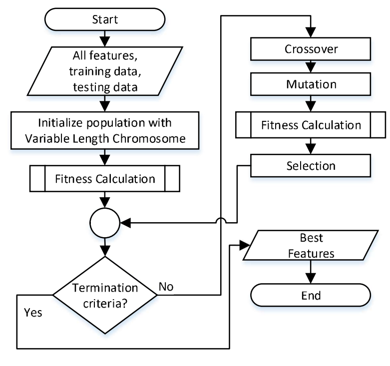

Fig. 7. VLCGA Flowchart

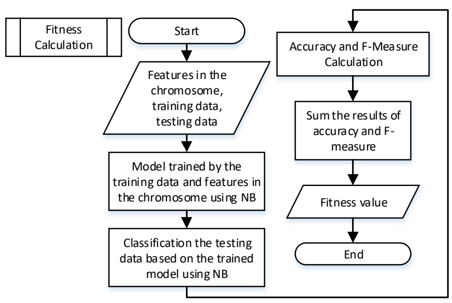

Fig. 8. Fitness Calculation Flowchart

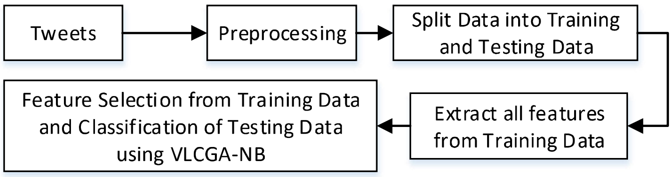

Fig. 9. Sentiment analysis steps

D. Naïve Bayes

In this study, the Naïve Bayes (NB) is utilized to classify the tweet's sentiment. The features were extracted by using unigrams after the preprocessing steps. This paper utilized the variable length chromosome, thus, the number of genes is different for each chromosome. The number of genes and features in each gene was randomly chosen by VLCGA from the extracted features.

The determination of each tweet's sentiment class is applied by training the NB classifier with the selected features in the chromosome of the VLCGA. NB is utilized in the fitness value calculation for each population in every generation of VLCGA (see Fig. 8).

III. Implementation and Results

The experiments were performed on a machine with Intel Core i7 CPU with 8 GB of DDR3 RAM. The OS was Windows. The code is developed in Python with the natural language toolkit NLTK and compiled using PyCharm.

The experiments consist of several steps, such as retrieve the manually classified Twitter data, preprocessing, training data and testing data separation, all feature extraction from training data, feature selection and classification using VLCGA (see Fig. 9)

For experimental settings, we use the following parameter for VLGCA-NB.

- Population size = 100
- Generation size = 50
- Crossover rate = 0.7
- Mutation rate = 0.3

The experiment conducted ten times and taken the average value of F-Measure, number of features, accuracy, and fitness value. The comparisons that we conducted are standard deviation comparison, average fitness comparison, average F-Measure for very positive sentiment, average F-Measure for positive sentiment, average F-Measure for neutral sentiment, average F-Measure for negative sentiment, average F-Measure for very negative sentiment, number of features comparison, and the accuracy comparison.

A. Average Fitness Comparison

The comparison of the average fitness done up to 50 generations. The population starts converging at the 27th generation with 2.9171 of average fitness as shown in Fig. 10.

B. Standard Deviation Comparison

We compare the standard deviation from 1 to 50 generation. From Fig. 11, the population starts converging at the 27th generation. The best standard deviation resulted is 0.10676. The standard deviation shows the fluctuation of the outcome. If the standard deviation is small, then outcomes are grouped near one another or near the mean. If the standard deviation is big, the outcomes are spread broadly and the fluctuation is high.

C. Precision and Recall

From Table III and IV, the number of correctly identified for Very Positive tweets by NB classification is really low, with 6.9% recall and the correctly classified of Very Positive tweets is only 15.4%. The results are very bad precision leads to 84.6% false positives for the Very Positive class. Although the resulted precision and recall by VLCGA-NB is also low, the average precision value for the Very Positive class is increased 126.7% and the average recall is increased 55% compared with the precision and recall of NB classification only.

The increase also occurred in the precision and recall results of Positive, Neutral, Negative, and Very Negative, except for the Neutral class precision. The NB classification

TABLE III. Precision Comparison

		Precision				
		Very Positive	Positive	Neutral	Negative	Very Negative
Method	Naïve Bayes	0.1538	0.4181	0.7936	0.5	0.2
	Average results					
	VLCGA-NB	0.3487	0.4651	0.7888	0.61433	0.58690
	Best results					
	VLCGA-NB	0.5555	0.5434	0.8050	0.8	0.75

TABLE IV. Recall Comparison

		Recall				
		Very Positive	Positive	Neutral	Negative	Very Negative
Method	Naïve Bayes	0.0689	0.3066	0.9124	0.2777	0.1666
	Average results					
	VLCGA-NB	0.1068	0.3293	0.9324	0.2472	0.3500
	Best results of features					
	VLCGA-NB	0.1724	0.3866	0.9519	0.3055	0.5

TABLE V. F-Measure Results, Number of Features, Accuracy And Fitness Value Comparison

		F-Measure					Number of Features	Accuracy	Fitness
		Very Positive	Positive	Neutral	Negative	Very Negative			
Method	Naïve Bayes	0.095238	0.353846	0.848883	0.357143	0.181818	1086	0.718	2.554928
	Average results								
	VLCGA-NB	0.162273	0.38371	0.854481	0.348081	0.430769	403	0.7378	2.917114
	Best results								
	VLCGA-NB	0.263158	0.442748	0.864103	0.4	0.6	188	0.752	3.088953

produced higher precision than the VLCGA-NB's average precision value. However, the best precision value of VLCGA-NB is still higher than NB.
The tweets are correctly identified as a Neutral class with high precision and recall by NB and VLCGA-NB. While other class produced very low values. This caused by the different total number of tweets for each class, where the number of neutral tweets is more than the other classes.

D. *Average F-Measure*

The average for Very Positive sentiment is 0.16227. Fig. 12 shows that the F-Measure of Very Positive sentiment increasing until the results start to converge on the 27th generation. The F-Measure of Positive sentiment seen in Fig. 13 increases at 0.38371 and converge at 27th generation. The chart of the Average F-Measure for Neutral sentiment seen in Fig. 14 is fluctuating and start to converge at 27th with the average value of 0.85448.

As seen in Fig. 15, the average F-Measure for Negative sentiment down drastically in the fourth generation. But start fluctuating until the 27th generation with the average value of 0.34808. The chart of the average F-Measure for Very Negative sentiment seen in Fig. 16 is fluctuating and start to converge at 20th with the average value of 0.430769.

From Table V, it shows that the F- Measure scores for very positive, positive, very negative, and negative are very low, and only neutral class has the high F-Measure score. This caused by the imbalance number of tweets for each sentiment. From Table I, the number of neutral tweets is more than the other sentiments.

E. *F-Measure, Number of Features, Accuracy and Fitness Value Results Comparison*

This research compares the Average and Best F-Measure of each sentiment for NB and VLCGA-NB. From Table V, it shows that the VLCGA-NB produce the higher F-Measure for all sentiments than NB. However, the average F-Measure for negative sentiment is lower than the NB. This may be caused by the imbalance number of tweets composition.

Moreover, Table V shows that VLCGA-NB produces the lower number of features, higher accuracy, and higher fitness value than NB. The comparison shows that the VLCGA-NB prove its ability to reduce the number of features and increase the accuracy.

IV. Conclusions

A combination of an algorithm with an optimization algorithm enhances the ability of a classifier. An optimization is performed to reduce the features in the sentiment analysis problem. In this research, NB is used to produce a trained model which will be used to classify the testing data. The classification result is then used to calculate the fitness value for each population in each generation of VLCGA. Finally, the conventional NB and VLCGA-NB are compared by its F-Measure of each sentiment, a number of features, accuracy, and fitness value for tweets related to self-driving cars.

The experiments show that overall the VLCGA-NB produces the higher value for the Best F-Measure results of each sentiment, lower number of features, higher accuracy, and higher fitness value than NB. However, NB outperforms the VLCGA-NB for the Average F-Measure in the negative sentiment. The lower accuracy in VLCGA-NB for the negative sentiment might be caused by imbalance number of tweets for each sentiment.

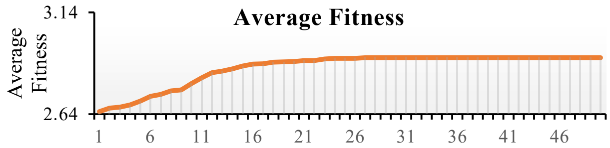

Fig. 10. Average fitness comparison between generation

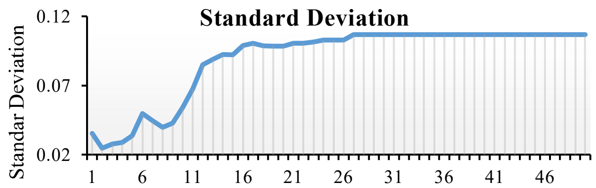

Fig. 11. Standard deviation comparison between generation

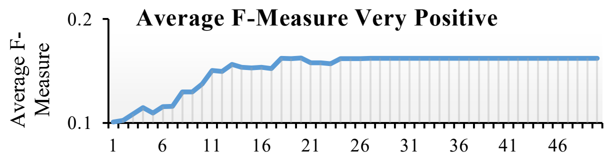

Fig. 12. Average F-Measure for very positive sentiment between generation

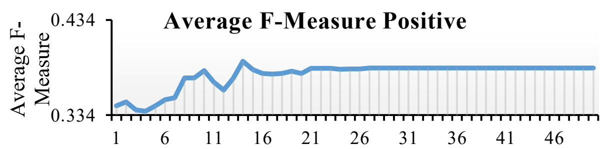

Fig. 13. Average F-Measure for positive sentiment between generation

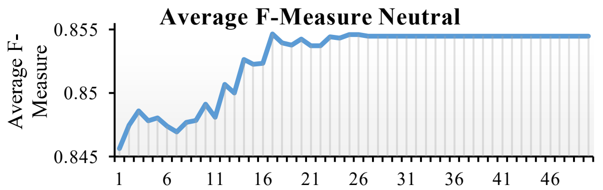

Fig. 14. Average F-Measure for Neutral sentiment between generation

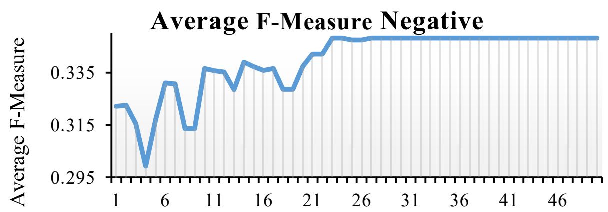

Fig. 15. Average F-Measure for Negative sentiment between generation

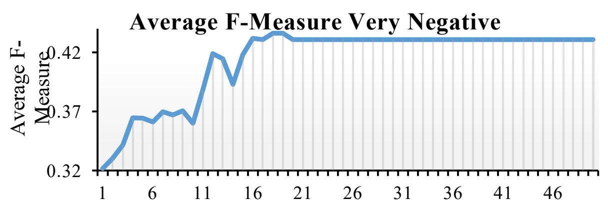

Fig. 16. Average F-Measure for very negative sentiment between generation

For future research, the accuracy can be increased by implementing an algorithm to fix the class imbalance problem. As shown in Table I, the tweets composition for each class is not balanced. This will affect the fitness value as the model that generated from the training stage will focus on the class with the higher quantity.

References

[1] Aslam S. Twitter by the Numbers: Stats, Demographics & Fun Facts. 2018 [cited 2018 Jul 8]. Available from: https://www.omnicoreagency.com/twitter-statistics/

[2] Fatyanosa TN, Bachtiar FA. *Classification method comparison on Indonesian social media sentiment analysis*. In: 2017 International Conference on Sustainable Information Engineering and Technology (SIET). IEEE; 2017. p. 310–5.

[3] Song K, Yao T, Ling Q, Mei T. Boosting image sentiment analysis with visual attention. *Neurocomputing*. 2018 Jun;

[4] Yoo S, Song J, Jeong O. Social media contents based sentiment analysis and prediction system. *Expert Systems with Applications*. 2018 Sep;105:102–11.

[5] Qiu J, Liu C, Li Y, Lin Z. Leveraging sentiment analysis at the aspects level to predict ratings of reviews. *Information Sciences*. 2018 Jul;451–452:295–309.

[6] Krouska A, Troussas C, Virvou M. *The effect of preprocessing techniques on Twitter sentiment analysis*. In: 2016 7th International Conference on Information, Intelligence, Systems & Applications (IISA). IEEE; 2016. p. 1–5.

[7] Kateb F, Kalita J. Classifying Short Text in Social Media: Twitter as Case Study. *International Journal of Computer Applications*. 2015 Feb 18;111(9):1–12.

[8] Othman R, Belkaroui R, Faiz R. Extracting Product Features for Opinion Mining Using Public Conversations in Twitter. *Procedia Computer Science*. 2017;112:927–35.

[9] FICAMOS P, LIU Y. A Topic based Approach for Sentiment Analysis on Twitter Data. *International Journal of Advanced Computer Science and Applications*. 2016;7(12).

[10] Wu Y-L, Tang C-Y, Hor M-K, Wu P-F. Feature selection using genetic algorithm and cluster validation. *Expert Systems with Applications*. 2011 Mar;38(3):2727–32.

[11] Ferriyan A, Thamrin AH, Takeda K, Murai J. *Feature selection using genetic algorithm to improve classification in network intrusion detection system*. In: 2017 International Electronics Symposium on Knowledge Creation and Intelligent Computing (IES-KCIC). IEEE; 2017. p. 46–9.

[12] Nguyen N-T, Lee H-H, Kwon J-M. Optimal feature selection using genetic algorithm for mechanical fault detection of induction motor. *Journal of Mechanical Science and Technology*. 2008 Mar 31;22(3):490–6.

[13] Selvaperumal P, Suruliandi A. *A short message classification algorithm for tweet classification*. In: 2014 International Conference on Recent Trends in Information Technology. IEEE; 2014. p. 1–3.

[14] Naradhipa AR, Purwarianti A. *Sentiment classification for Indonesian message in social media*. In: 2012 International Conference on Cloud Computing and Social Networking (ICCCSN). Bandung, Indonesia: IEEE; 2012. p. 1–5.

[15] Lunando E, Purwarianti A. *Indonesian social media sentiment analysis with sarcasm detection*. In: 2013 International Conference on Advanced Computer Science and Information Systems (ICACSIS). IEEE; 2013. p. 195–8.

[16] Parveen H, Pandey S. *Sentiment analysis on Twitter Data-set using Naive Bayes algorithm*. In: 2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT). IEEE; 2016. p. 416–9.

[17] Masrani M, Poornalatha G. Twitter Sentiment Analysis Using a Modified Naïve Bayes Algorithm. In 2018. p. 171–81.

[18] Goel A, Gautam J, Kumar S. *Real time sentiment analysis of tweets using Naive Bayes*. In: 2016 2nd International Conference on Next Generation Computing Technologies (NGCT). IEEE; 2016. p. 257–61.

[19] Pawar SN, Bichkar RS. Genetic algorithm with variable length chromosomes for network intrusion detection. *International Journal of Automation and Computing*. 2015 Jun 9;12(3):337–42.

[20] Kim IY, de Weck OL. Variable chromosome length genetic algorithm for progressive refinement in topology optimization. *Structural and Multidisciplinary Optimization*. 2005 Jun 7;29(6):445–56.

[21] Ni J, Wang K, Huang H, Wu L, Luo C. *Robot path planning based on an improved genetic algorithm with variable length chromosome*. In: 2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD). IEEE; 2016. p. 145–9.

[22] Qiongbing Z, Lixin D. A new crossover mechanism for genetic algorithms with variable-length chromosomes for path optimization problems. *Expert Systems with Applications*. 2016 Oct;60:183–9.

[23] Figure Eight. Twitter sentiment analysis: Self-driving cars. 2015 [cited 2018 May 19]. Available from: https://www.figure-eight.com/data-for-everyone/

[24] Nordquist R. What Are Contractions in Grammar? List of Contractions. 2018 [cited 2018 May 18]. Available from: https://www.thoughtco.com/contractions-commonly-used-informal-english-1692651

[25] Manning CD, Raghavan P, Schütze H. The term vocabulary and postings lists. In: Introduction to Information Retrieval. Cambridge, United Kingdom: Cambridge University Press; 2008. p. 19–47.

[26] Mahmudy WF, Marian RM, Luong LHS. Real Coded Genetic Algorithms for Solving Flexible Job-Shop Scheduling Problem - Part II: Optimization. *Advanced Materials Research*. 2013 May;701:364–9.