OPEN ACCESS
J. OF INISTA, VOL. 5, NO. 1, PP.080-091, NOV 2022

ISSN : 2622-8106 (ONLINE)
DOI: 10.20895/INISTA.V5I1.838

Journal of Informatics, Information System, Software Engineering and Applications

Indonesian Sentiment Analysis towards MyPertamina Application Reviews by Utilizing Machine Learning Algorithms

Fiddin Yusfida A'la

Department of Informatics Engineering, Universitas Sebelas Maret
Surakarta, Indonesia

fiddin@staff.uns.ac.id

Accepted on 31-12-2022

Abstract

This paper is a report of experiment analysis on sentiment analysis in application review that explored the methods and the data. Application review contains a large amount of raw data that has been published by users in the form of text, image, audio, and video. The data can be converted into valuable information by using sentiment analysis. In this work, around 5000 Indonesian review in MyPertamina google play application are analyzed. The goal of this study was to investigate the effectiveness of using sentiment analysis to extract valuable insights from application reviews. Some techniques were applied during this work, such as data collection, pre-processing, feature extraction, TF-IDF text representation, machine learning modelling, and evaluation phase. The machine learning algorithms that we used are Linear Support Vector Classification (Linear SVC) and Multinomial Naïve Bayes (Multinomial NB). The result shows both machine learning models present good performance in this data. The accuracy of Multinomial NB reaches 95%, while Linear SVC obtains 96% of accuracy. The results of the experiment suggest that both Linear SVC and Multinomial NB are well-suited for sentiment analysis tasks on Indonesian language data. Future work could include expanding the dataset to include reviews from a broader range of applications, or evaluating the performance of additional machine learning algorithms. In addition, word cloud analysis also performed in this experiment. The word cloud shows that positive and negative sentiment present some popular words which appear inside the review. It would also be interesting to conduct a deeper analysis of the word cloud results to identify common themes and trends in the positive and negative sentiments expressed in the reviews.

Keywords: application review, big data, mypertamina, sentiment analysis.

I. INTRODUCTION

In the last decade, information and technology have redefined social norms. Mobile applications can be able to change our daily live and behavior [1]. There are a lot of applications in internet, it can be desktop applications, web-based applications, and mobile applications. However, mobile applications look simple, compact, and easy to access since smartphones has already become a part of our live. Using this technology, everything is simple. Simply by finding the appropriate apps and reading through user reviews and ratings.

In other views, as a business owner or product owner, listening to customer feedback is important to improve the experience of mobile apps [2]. Recently, rely on application review rating are not enough, comment from reviews also play a vital role to understand the user's perspective. Using sentiment analysis approaches, we can identify and determine if review indicates a positive, negative, or neutral emotion. It leads in understanding customer feedback and mostly used by several companies to analyze application review. It also helps to find out the underlying sentiment in a text.

Natural language processing (NLP) theoretically defines as computational procedures for analyzing and reflecting naturally occurring texts, at one or more levels of linguistic analysis for the purpose of achieving human-like language processing for a range of tasks or applications [3]. One of the main goals of NLP is to allow computers to understand, interpret, and generate human language, which can be challenging due to the complexity and variability of natural language. In addition, NLP can be called as a tool with which a document can be processed to find positive, negative, or neutral sentiments. It can be useful in identifying trend and user's sentiments towards a product or services. As a result, business objective can be modified to address customer's concerns. There is a lot of application to ease the end user, such as MyPertamina. MyPertamina is a platform for online financial services provided by Pertamina Indonesia. At Pertamina's public fueling facilities, this application is used to process non-cash fuel oil payments [4]. In the context of this research, NLP was used to analyze the sentiment of reviews left by MyPertamina users in order to identify common complaints and areas for improvement. However, many MyPertamina users complain about the application. The purpose of this research is to analyze and elaborate the application review from end user perspective. By understanding the sentiments of users, the development team behind MyPertamina can make informed decisions about how to improve the application and better meet the needs of its users. The results of the NLP analysis may also be useful for marketing and customer service teams, as they can use the insights gained to address user concerns and improve the overall user experience.

The contribution of this paper includes (1) use two machine learning algorithms for sentiment analysis on 5000 MyPertamina Indonesian reviews, (2) use text representation namely Term Frequency – inverse Document Frequency (TF-IDF), (3) the supervised machine learning algorithms such as Multinomial Naïve Bayes and Linear Support Vector Classification was implemented and evaluated using accuracy as performance criteria, (4) word cloud data visualization also implemented and analyzed to extract the most popular word inside the application review.

Furthermore, this paper is organized into five sections. Section 2 demonstrates previous related work or literature review related to implementation of sentiment analysis on Indonesian application review. Section 3 introduces experimental research methodology. Section 4 depicts the experiment result and discussion. The study's final statements and recommendations for further work are included in Section 5.

II. LITERATURE REVIEW

There are some previous works related to sentiment analysis, application review, machine learning, and big data. A comprehensive analysis of twitter data using machine learning has been discussed by Kawade [5]. The data that has been collected is 5000 tweets. The result shows that even limited character such as twitter data can be used for analyzing the content of the user's perspective. Research by Handani [6], tried to analyze Go-Jek application review. Go-Jek is an online transportation services application. In their work, Naïve Bayes Classifier has been used to build data model. The result shows this algorithm can be used for text mining especially for analyzing application review. A study for analyzing mobile telecommunication service namely by.U has been done by Fransiska [7]. This research uses google play scrapper library. From the collected reviews, they labelled the data manually, for score 1 and 2 converted to negative sentiment, while score 4 and 5 converted to positive sentiment. Based on their work, score 3 was not included to the research since it has less informative. The document (reviews) has been transformed into vector by using TF-IDF technique. The result shows that the TF-IDF and Support Vector Machine (SVM) methods can be applied to the classification process with good measurement results. However, further analyze according to Indonesia application review need to be done. This research tries to analyze the MyPertamina application review using word cloud method to know the most popular word in positive and negative score. In addition, this research also compares two machine learning which powerful algorithms according to sentiment analysis such as Linear SVC and Multinomial NB.

III. RESEARCH METHOD

Fig. 1. presents the experimental design. The experiment divided into some parts such as data collection, pre-processing phase, feature extraction, word evaluation using TF-IDF, build two classification models based on training data and evaluation phase. However, inside the pre-processing phase, there are some techniques to be applied to the data such as data normalization, case folding, data filtering, character repetition removal, tokenization, and stop words removal. The detail of every phase will be elaborated in the next section.

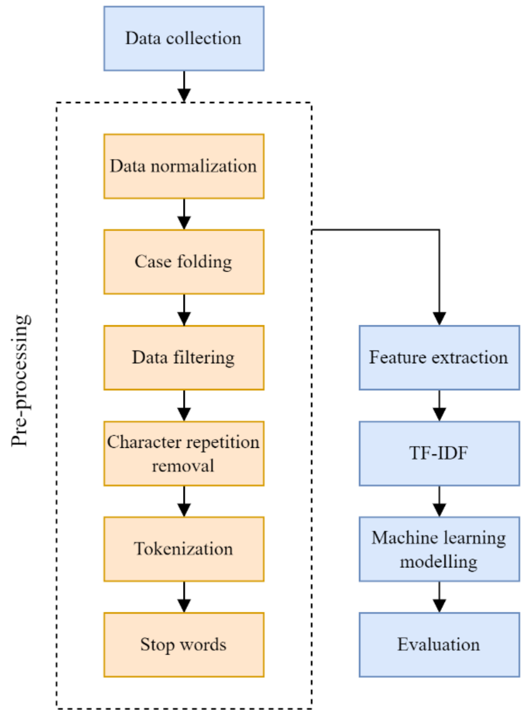

Fig. 1. Experimental Design

A. *Data Collection and Analysis*

In this part, some application reviews were collected. In this phase, we obtained 5000 application reviews. The library that we used is google play scrapper. From this work we collected data based on its attributes such as review id, username, user image, content, score, thumb up count, review created version, published time, reply content, and replied at. However, in this work, we only use content and score as parameter since it contains more information to classify the review. The sample of the reviews is shown in Table I.

TABLE I
REVIEW SAMPLE

No	Review Id	Username	User Image	Content	Score	Thumbs Up Count	Review Created Version	at	Reply Content	Replied At
1	ab595d48-254a-40e5-a1a4-	Gilang ramadhan	https://play-lh.googleusercontent.com/a/AItbvm...	Rumit	1	0	3.6.3	2022-07-18 12:11:44	None	NaT

	d41e9b6ead1f									
2	78fae9c6-8ae0-41d8-87c7-311b630ca378	bismi eko	https://play-lh.googleusercontent.com/a-/AFdZu...	Jaringan sering error	1	0	None	2022-07-18 12:10:55	None	NaT
3	a4319c2f-55b4-43fb-a0d4-4a8010baaa01	Rezky Rinaldi	https://play-lh.googleusercontent.com/a-/AFdZu.	Metode pembayarannya di perbanyak	3	0	3.6.3	2022-07-18 12:04:28	None	NaT

The graph of the data and its score is shown in Fig. 2. It shows that 93% of data are labelled as score 1, while the least score is 4 (0.4%). Majority of end user perspective about MyPertamina gives bad review. The score 1 and score 5 have significant differences.

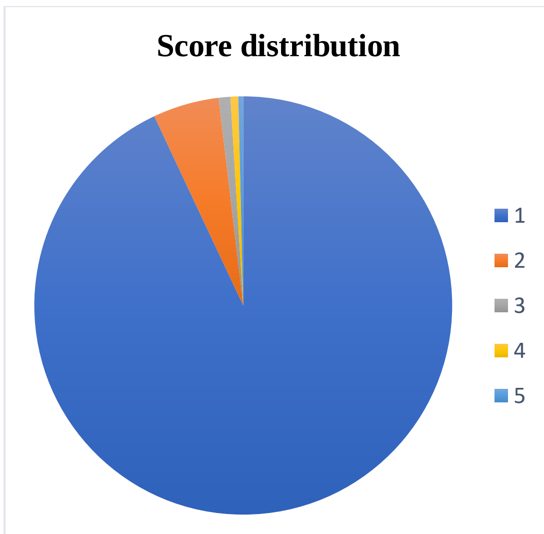

Fig. 2. Data Distribution

The detail of the score is sorted by its frequency, the data is shown in Table II. From this figure, score 1 appears 4650 times, score 5 (256 times), score 2 (45 times), mid score 3 appears 30 times, and the least frequency comes from score 4 (19 times).

TABLE II
REVIEW SORTED BY ITS SCORE

Score	Frequency
1	4650
5	256
2	45
3	30
4	19

In this work, we only use two attributes such as content and score. We use these two attributes since the other attributes give less information about the review. The sample of the data is shown in Table III.

TABLE III
SAMPLE OF THE REVIEW AND ITS SCORE

Content	**Score**
Rumit	1
Jaringan sering error	1
Metode pembayarannya di perbanyak	3

However, in this work, we only use review which has score 1 and 5. After obtaining score 1 and 5 we convert those values into 0 and 1 respectively. Score 0 contains 4650 reviews and score 1 contains 256 reviews.

B. *Pre-processing*

1) *Data Normalization:*

After obtaining data, the next step is data normalization. The purpose of this phase is to normalize the review since some reviews contains slang words or clean the data from the noise [8]. This work has been done using Indonesian colloquial words collection [9]. The result is shown in Table IV.

TABLE IV
DATA NORMALIZATION RESULT

Before Normalization	**After Normalization**
'aplikasi burik, mau daftar **aj** susah'	'aplikasi burik , mau daftar **saja** susah'
'Cuma GABUT **ngasih** Bintang satu 🗿'	'Cuma GABUT **mengasih** Bintang satu 🗿'
'Aplikasi gak jelas,,,pemerintah harusnya membuat aplikasi **yg** lebih baik dari ini,, Masyarakat di suruh bayar lewat aplikasi sementara aplikasinya error **gak** jelas,,,'	'Aplikasi enggak jelas , , , pemerintah harusnya membuat aplikasi **yang** lebih baik dari ini , , Masyarakat di suruh bayar lewat aplikasi sementara aplikasinya error **enggak** jelas , , ,'

2) *Case Folding:*

All letters are transformed into lowercase [10]. The sample of the text before and after case folding is shown in Table V.

TABLE V
CASE FOLDING RESULT

Before Case Folding	**After Case Folding**
'aplikasi burik , mau daftar saja susah'	'aplikasi burik , mau daftar saja susah'
'Cuma GABUT mengasih Bintang satu 🗿'	'cuma gabut mengasih bintang satu 🗿'
'Aplikasi enggak jelas , , , pemerintah harusnya membuat aplikasi yang lebih baik dari ini , , Masyarakat di suruh bayar lewat aplikasi sementara aplikasinya error enggak jelas , , ,'	'aplikasi enggak jelas , , , pemerintah harusnya membuat aplikasi yang lebih baik dari ini , , masyarakat di suruh bayar lewat aplikasi sementara aplikasinya error enggak jelas , , ,'

3) *Data Filtering:*

The purpose of this step is taking important word from less important words [11]. In other words, only saving important words and discarding less important words. The sample of filtering result is shown in Table VI.

TABLE VI
DATA FILTERING RESULT

Before Data Filtering	After Data Filtering	Removed Parts
'aplikasi burik , mau daftar saja susah'	'aplikasi burik mau daftar saja susah'	,
'cuma gabut mengasih bintang satu 🗿'	'cuma gabut mengasih bintang satu '	🗿
'aplikasi enggak jelas , , , pemerintah harusnya membuat aplikasi yang lebih baik dari ini , , masyarakat di suruh bayar lewat aplikasi sementara aplikasinya error enggak jelas , , ,'	'aplikasi enggak jelas pemerintah harusnya membuat aplikasi yang lebih baik dari ini masyarakat di suruh bayar lewat aplikasi sementara aplikasinya error enggak jelas '	, space

4) *Character Repetition Removal:*

In this step, some repetition characters have been removed from the sentence. The sample of the character removal phase is shown in Table VII.

TABLE VII
CHARACTER REMOVAL RESULT

Before Character Removal	After Character Removal
b b b sampai	b b b sampai

5) *Tokenization:*

Tokenization is a step which separating a piece of text into smaller unit called token. The tokens contain words, symbol, punctuation marks, numbers, and other important entities [12]. The detail reviews after tokenization phase are shown in Table VIII.

TABLE VIII
TOKENIZATION RESULT

Sentence	Token
jaringan sering error	['jaringan', 'sering', 'error']
Ini sekelas bumn buat aplikasi banyak errornya enggak bisa login lewat web juga enggak bisa buruk banget	['ini', 'sekelas', 'bumn', 'buat', 'aplikasi', 'banyak', 'errornya', 'enggak', 'bisa', 'login', 'lewat', 'web', 'juga', 'enggak', 'bisa', 'buruk', 'banget']
gak bisa request lagu	['gak', 'bisa', 'request', 'lagu']

6) *Stop Word Removal:*

The purpose of this stage is to exclude words that appear frequently across all of the corpus's documents [13]. In this work, we use Indonesian corpus provided by Natural Language Toolkit (NLTK) library. Stop word removal result is shown in Table IX.

TABLE IX
STOP WORD REMOVAL RESULT

Before Stop Word Removal	After Stop Word Removal
['jaringan', 'sering', 'error']	jaringan error
['ini', 'sekelas', 'bumn', 'buat', 'aplikasi', 'banyak', 'errornya', 'enggak', 'bisa', 'login', 'lewat', 'web', 'juga', 'enggak', 'bisa', 'buruk', 'banget']	sekelas bumn aplikasi errornya login web buruk banget
['gak', 'bisa', 'request', 'lagu']	gak request lagu

C. *Feature Extraction and Data Weighting*

Feature extraction technique is applied to the data. One of the quickest and most effective text mining techniques is the TF-IDF approach [14]. Output of this step is extracted features and its weight and ready to be trained by machine learning algorithms.

D. *Machine Learning Modelling*

Machine learning algorithms that we used are Linear Support Vector Classification (Linear SVC) and Multinomial Naïve Bayes (Multinomial NB). In this work we only use default parameters from scikit-learn without any adjustments. Output of this step is two classification models namely Linear SVC and Multinomial NB model.

E. *Evaluation*

Evaluation step aim is to measure how accurate the model prediction. Output of this step is percentage of accuracy for every classification model (Linear SVC and Multinomial NB). The equation of the accuracy is shown in Eq. 1. And the confusion matrix is shown in Table X.

TABLE X
CONFUSION MATRIX

		Actual values	
		Positive (1)	Negative (0)
Predicted values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Annotation:
TP : true positive
FP : false positive
FN : false negative
TN : true negative

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

IV. RESULTS AND DISCUSSION

In this section, extensive experiment results are reported. For instance, the bag of words sample is shown in Table XI.

TABLE XII
BAG OF WORDS SAMPLE

	aplikasi	bagus	bensin	error	isi	jaringan	mantap	pakai	pertamina	ribet	class
Doc0	0	0	0	1	0	1	0	0	0	0	Neg
Doc1	1	0	1	0	1	0	0	1	0	1	Neg
Doc2	0	0	0	0	0	0	0	0	0	1	Neg
Doc3	0	0	0	0	0	0	1	0	1	0	Pos
Doc4	0	1	0	0	0	0	0	0	0	0	Pos

A. *Naïve Bayes for Sentiment Analysis*

The Naive Bayes equation for sentiment analysis is typically used to predict the probability that a given text document belongs to a particular class or category, such as "positive" or "negative" sentiment. First, calculate the prior probability for each class. The sample are shown in E.q. 2 and E.q. 3.

$$P(Neg) = \frac{3}{5} = 0.6 \quad (2)$$
$$P(Pos) = \frac{2}{5} = 0.4 \quad (3)$$

Calculate the likelihood of each feature (i.e., aplikasi, bagus, bensin, etc.) and repeat for all words. This can be done by counting the number of occurrences of each feature in each class, and dividing by the total number of observations in that class. Laplace smoothing is a method that is used to prevent zero probabilities when training a Naive Bayes model. This is done by adding a small constant, often 1, to each count. The sample are shown in E.q. 4 and E.q.5.

$$P(aplikasi|Neg) = \frac{1+1}{8+10} = 0.11 \quad (4)$$
$$P(aplikasi|Pos) = \frac{0+1}{3+10} = 0.08 \quad (5)$$

Finally, use these prior probabilities and likelihoods to classify new observations by calculating the posterior probability for each class. In this sample, we test "aplikasi error" for sample. Based on E.q. 6 and E.q.7, the class with the highest probability will be the predicted class for the new observation.

$$P(aplikasi\ error|Pos) = P(Pos) * P(aplikasi|Pos) * P(error|Pos) = 0.4 * 0.08 * 0.08 = 0{,}0026 \quad (6)$$

$$P(aplikasi\ error|Neg) = P(Neg) * P(aplikasi|Neg) * P(error|Neg) = 0.6 * 0.11 * 0.11 = 0{,}0073 \quad (7)$$

B. *Support Vector Classification for Sentiment Analysis*

In sentiment analysis, SVM can be used to separate text documents into different sentiment classes, such as "positive" and "negative.". In this example, we are using the sample sentence "aplikasi error" to test the sample data provided. The first step would be to tokenize the sentence, which means breaking it up into individual words. In this case, the sentence "aplikasi error" would be tokenized into the words "aplikasi" and "error". Then we would need to create a vector with the same number of columns as the training data, where each column represents a word. For each word in the sentence, we would check if it appears in the training data and set the corresponding value in the vector to 1 if it does, and 0 if it doesn't. For this example, the vector would be [1,0,0,1,0,0,0,0,0,0]. The example of boundary is shown in E.q. 8, where x_1, x_2, x_3, ... x_10 are the input features.

$$-2x\_1 - 3x\_2 + x\_3 - 2x\_4 + x\_5 - x\_6 - x\_7 + x\_8 + x\_9 - x\_10 = 0 \quad (8)$$

In this case, the weight vector of the model would be [-2, -3, 1, -2, 1, -1, -1, 1, 1, -1] and the bias term b would be 0. Now, we can plug in the input vector [1,0,0,1,0,0,0,0,0,0] into the E.q.9:

$$-21 - 30 + 10 - 21 + 00 - 00 - 00 + 00 + 00 - 00 = -2 \quad (9)$$

This value can be used to calculate the probability of the input vector belonging to the positive class using the formula (E.q. 10)

$$P(y = 1|x) = 1/(1 + exp(-decision\_function)) \quad (10)$$

where decision_function is the value of the equation obtained above, which is -2. (E.q. 11)

$$P(y = 1|x) = 1/(1 + exp(2)) = 0.11920292202211757 \quad (11)$$

It means that there is 0.11920292202211757 probability that input vector belongs to the positive class or 0.8807970779778823 probability that the input vector belongs to the negative class.

C. *Word Cloud Analysis*

Word cloud is one of data visualization method which based on word frequency [15][16]. Word cloud result is shown in Fig. 3. and Fig. 4. From these figures we can conclude that in positive reviews, there are a lot of praises related to MyPertamina apps. However, based on negative word cloud shows that some aspects that make end user gives less score in their review.

Fig. 3. Positive Word Cloud

Fig. 4. Negative Word Cloud

D. *Popular Word*

Popular word is shown in Fig. 5. and Fig. 6., the sample of popular words in positive sentiment are "bagus", "mantap", "membantu". On the other hand, the most popular words in negative sentiment are "ribet", "susah", "bug". Based on this result, the positive review says about their satisfaction while using the MyPertamina application, while the negative review says that some aspect can be improved for the related stakeholder. The popular words can be more specific in order to fasten the next version of the application and make the application more robust.

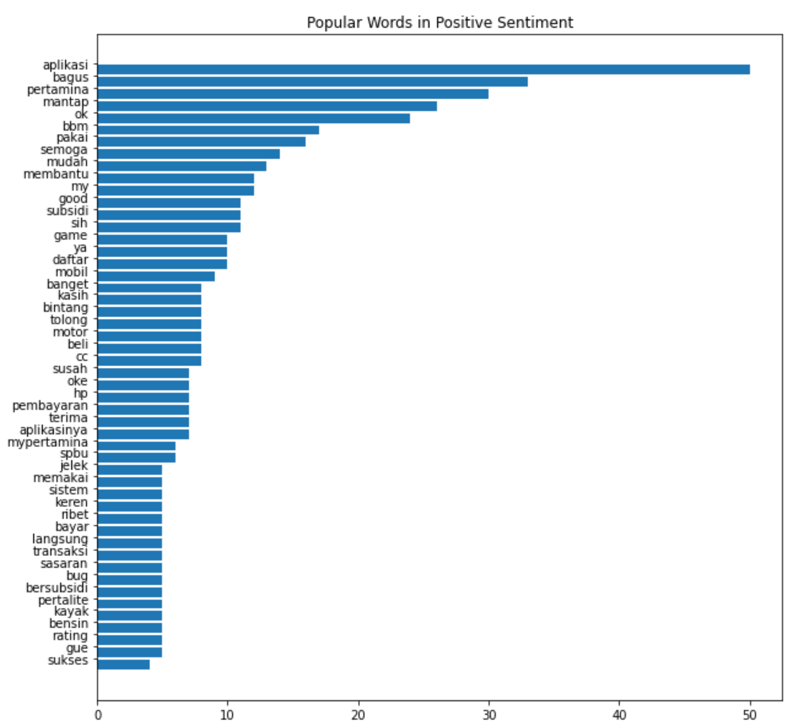

Fig. 5. Popular Words in Positive Review

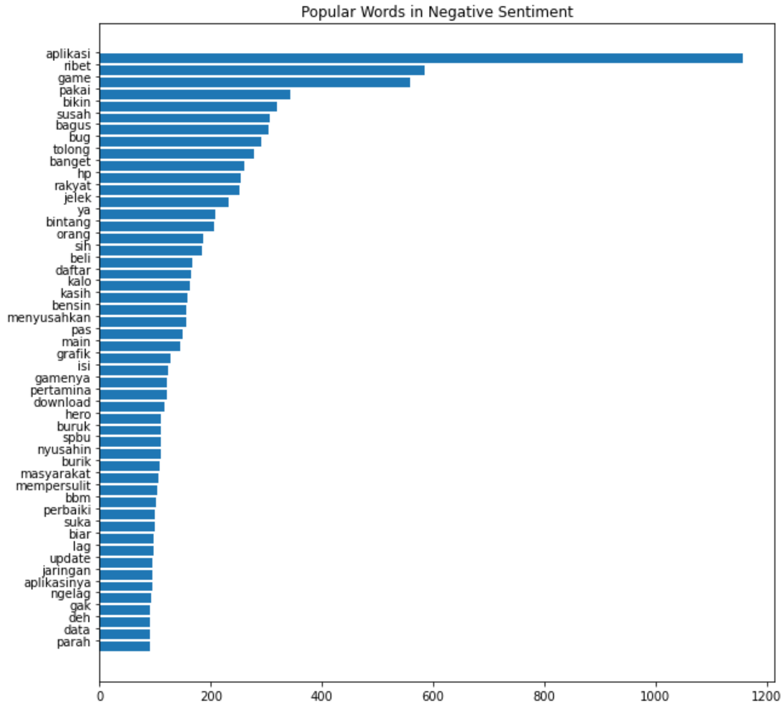

Fig. 6. Popular Words in Negative Review

E. *Performance Analysis*

Result for Multinomial Naïve Bayes and Support Vector Classifier are shown in Fig. 7. and Fig. 8. respectively. Multinomial NB shows that false negative frequency is 50, while true negative frequency is 932. Furthermore, in Linear SVC result shows that there are 36 data labelled as false negative and 14 data are detected as true positive. In addition, for Linear SVC performance demonstrates that there are 928 are true positive and 4 data detected as false positive. Overall evaluation for these two machine learning models is the accuracy of Multinomial NB is 95% and Support Vector Classification is 96%. The model that has been built using Linear SVC is slightly higher than Multinomial NB. The calculation is shown in E.q. 10 and E.q.11

$$NB\ accuracy = \frac{TN+TP}{TP+FP+TN+FN} = \frac{932+0}{0+50+932+0} = \frac{932}{982} = 0.95 \quad (10)$$

$$SVC\ accuracy = \frac{TN+TP}{TP+FP+TN+FN} = \frac{928+14}{0+50+932+0} = \frac{932}{982} = 0.96 \quad (11)$$

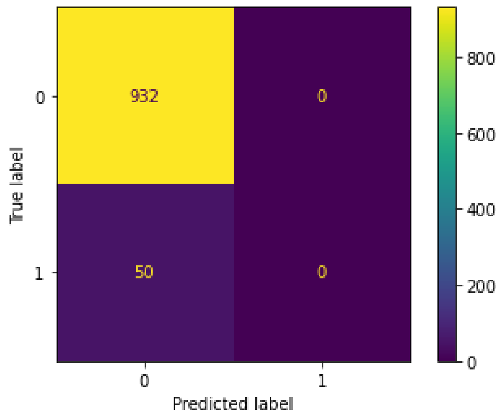

Fig. 7. Multinomial NB Result

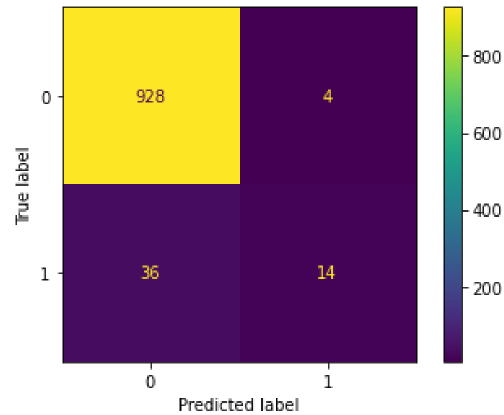

Fig. 8. Linear SVC Result

V. Conclusion

This paper has presented analysis of application review from google play using sentiment approach. In this research we use 5000 Indonesian application reviews. Some pre-processing phase such as data normalization, data filtering, character repetition removal, tokenization, stop words removal were applied. We also used TF-IDF to perform sentiment analysis. For machine learning modelling we used two algorithms namely Multinomial Naïve Bayes and Linear Support Vector Classification. Based on the research experiment, Linear SVC obtained 96% of accuracy and Multinomial NB obtained 95% of accuracy. It shows a robust classification model for predicting the Indonesian application review sentiment. In addition, based on word cloud analysis, both two classess show popular word. Some popular words in positive sentiment (score 5) are frequently typed by user such as "bagus", "mantap, "ok", while popular words in negative sentiment (score 1) such as "ribet", "susah", "bug". For future works, other review attributes can be included to the research such as Thumbs Up Count, Review Created Version, and add more machine learning algorithms.

REFERENCES

[1] A. Al-Subaihin *et al.*, "App store mining and analysis (keynote)," in *3rd International Workshop on Software Development Lifecycle for Mobile, DeMobile 2015 - Proceedings*, Aug. 2015, pp. 1–2. doi: 10.1145/2804345.2804346.

[2] M. Khalid, M. Asif, and U. Shehzaib, "Towards Improving the Quality of Mobile App Reviews," *International Journal of Information Technology and Computer Science*, vol. 7, no. 10, pp. 35–41, Sep. 2015, doi: 10.5815/ijitcs.2015.10.05.

[3] S. R. Joseph, H. Hlomani, K. Letsholo, F. Kaniwa, and K. Sedimo, "Natural Language Processing: A Review," 2016. [Online]. Available: http://www.euroasiapub.org

[4] MyPertamina, "MyPertamina," *MyPertamina*, 2021. https://mypertamina.id (accessed Sep. 13, 2022).

[5] D. R. Kawade and Dr. K. S. Oza, "Sentiment Analysis: Machine Learning Approach," *International Journal of Engineering and Technology*, vol. 9, no. 3, pp. 2183–2186, Jun. 2017, doi: 10.21817/ijet/2017/v9i3/1709030151.

[6] S. Wahyu Handani, D. Intan Surya Saputra, Hasirun, R. Mega Arino, and G. Fiza Asyrofi Ramadhan, "Sentiment analysis for go-jek on google play store," in *Journal of Physics: Conference Series*, Apr. 2019, vol. 1196, no. 1. doi: 10.1088/1742-6596/1196/1/012032.

[7] S. Fransiska and A. Irham Gufroni, "Sentiment Analysis Provider by.U on Google Play Store Reviews with TF-IDF and Support Vector Machine (SVM) Method," *Scientific Journal of Informatics*, vol. 7, no. 2, pp. 2407–7658, 2020, [Online]. Available: http://journal.unnes.ac.id/nju/index.php/sji

[8] M. Javed and S. Kamal, "Normalization of Unstructured and Informal Text in Sentiment Analysis," 2018. [Online]. Available: www.ijacsa.thesai.org

[9] N. A. Salsabila, Y. A. Winatmoko, A. A. Septiandri, and A. Jamal, "Colloquial Indonesian Lexicon," in *2018 International Conference on Asian Language Processing (IALP)*, 2018, pp. 226–229.

[10] Y. HaCohen-Kerner, D. Miller, and Y. Yigal, "The influence of preprocessing on text classification using a bag-of-words representation," *PLoS One*, vol. 15, no. 5, May 2020, doi: 10.1371/journal.pone.0232525.

[11] D. AbishekB, M. Arul Stephen, and D. Vijayalakshmi, "Data filtering and visualization for sentiment analysis of e-commerce websites," 2021. [Online]. Available: https://ssrn.com/abstract=3882733

[12] G. Gupta and S. Malhotra, "Text Document Tokenization for Word Frequency Count using Rapid Miner (Taking Resume as an Example)." [Online]. Available: https://www.researchgate.net/publication/339527155

[13] D. J. Ladani and N. P. Desai, "Stopword Identification and Removal Techniques on TC and IR applications: A Survey," in *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*, Mar. 2020, pp. 466–472. doi: 10.1109/ICACCS48705.2020.9074166.

[14] W. B. Trihanto, R. Arifudin, and A. Muslim, "Information Retrieval System for Determining the Title of Journal Trends in Indonesian Language Using TF-IDF and Naïve Bayes Classifier," *Scientific Journal of Informatics*, vol. 4, no. 2, pp. 2407–7658, 2017, [Online]. Available: http://journal.unnes.ac.id/nju/index.php/sji

[15] C. Singh, "Word cloud analysis of the BJGP," *British Journal of General Practice*, pp. 148–148, 2012.

[16] F. Y. A'la, Hartatik, N. Firdaus, M. A. Safi'ie and B. K. Riasti, "A Comprehensive Analysis of Twitter Data: A Case Study of Tourism in Indonesia," 2022 1st International Conference on Smart Technology, Applied Informatics, and Engineering (APICS), Surakarta, Indonesia, 2022, pp. 85-89, doi: 10.1109/APICS56469.2022.9918757.