ISSN 2278-3091

Volume 9, No.1, January – February 2020

International Journal of Advanced Trends in Computer Science and Engineering

Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse90912020.pdf

https://doi.org/10.30534/ijatcse/2020/90912020

Classification of User Comment Using Word2vec and SVM Classifier

Rafly Indra Kurnia[1], Yoshua Daniel Tangkuman[2], Abba Suganda Girsang [3]

[1] Computer Science Department, BINUS Graduate Program-Master of Computer Science, Bina Nusantara University, Jakarta, Indonesia 11480, rafly.kurnia@binus.ac.id

[2] Computer Science Department, BINUS Graduate Program-Master of Computer Science, Bina Nusantara University, Jakarta, Indonesia 11480, yoshua.tangkuman@binus.ac.id

[3] Computer Science Department, BINUS Graduate Program-Master of Computer Science, Bina Nusantara University, Jakarta, Indonesia 11480, agirsang@binus.edu

ABSTRACT

Social media provides data that can be used for text classification, but these social media do not have a rating system. Comments taken from social media such as Twitter, does not provide a rating system which can help to classify comments based on their rating score. The goal of this study is to build a comment classification model using Word2vec and SVM classifier that can classify comments based on a rating scale from 1-5. The training data will be taken from user comments from play.google.com website in which each comment already has a rating. The purpose of the model is to classify comments from social media about mobile network applications. Comment classification is performed in order to help these businesses, as well as the users, to know the overall satisfaction of users who use these applications. The best F1 score obtained from this research using class elimination and stop word removal is 0.795.

Key words : multiclass, SVM classifier, text classification, Word2vec.

1. INTRODUCTION

Provider or mobile network operator is a business entity that provides wireless communication services to end-users who use services including radio spectrum allocation and wireless network infrastructure. The provider also provides services for end-users to gain internet access either through the use of mobile sim cards or through wireless connections (Wi-Fi). Nowadays, almost every provider in Indonesia has its own application in the Google Play Store. With the applications found in the Google Play Store, users can now conduct reviews based on the internet speed of provider, UI / UX of the applications, signal stability and other factors in the review column of the Google Play Store feature. Reviews conducted on the Google Play Store enables users to give their opinion by leaving a comment on the comment section and also rate the application on a rating scale of 1-5 stars, in which 1 star is the lowest possible rating and 5 stars being the highest possible rating. By taking the average of the ratings given by all users, the applications will then have their rating score and can be ranked according to their score. Users can leave comments and give their rating only if the user has a registered account on the Google Play Store. This step is done to ensure that the comments and ratings can be accounted for by the user who made a comment on an application provider.

Social commerce is a subset of electronic commerce that involves online interactions between users having similar interests by the use of social media and other online media that supports social interaction [1]. The development of web 2.0 and social networking sites led to the development of social commerce [2]. Social commerce benefits from these medias by utilizing user interactions as well as user feedback for promoting products and services. The user contributions through comments and feedback can affect the desire of users to buy or sell a product or service [3]. Therefore, the need for information extraction from texts, including user comments, have also increased [4]. An increasing number of internet users in a country will also increase the utilization of this phenomenon through online commerce [5]. Satisfaction and loyalty of customers who buys or sells products and online services are expected to be a factor for the assessment of online business success [6].

Increased use of online-based software such as social software (SSW) has made it possible to support users interacting digitally in real-time when making transactions for a product [7]. The use of social media and technology appropriate for aspects of shopping has become a major problem for retailers and related businesses [8]. The number of factors that influence the growth of social commerce is due to the increase of popularity of social media, network notification, social search and online collaborative shopping tools [9].

The mobile network industry is also affected by the use of social commerce. The applications and services offered by companies in the mobile network industry are often commented on by users in social media platforms with user-generated content such as Twitter, Facebook, and Instagram. Another platform also used for social commerce is Google Play Store in which users can give comments and rate the applications that they want to review.

Reviews that can be collected are not only limited to applications that provide comment features such as the Google Play Store. Social media like Twitter and Facebook are also often used by users to give their opinions about many subjects, including the quality of a mobile network operator. However, since these social media do not have a rating system, companies being commented on cannot immediately discern the sentiment of these comments. As a result, many companies and businesses have a strong demand for sentiment classification. This study entitled "Classification of Provider User Comment in Google Play Using Word2vec and SVM Algorithm" will discuss the use of Word2vec and SVM algorithm to build a comment classification model for sentiment analysis based on data from provider application MyTelkomsel.

2. RELATED WORK

This section overviews the existing works related to sentiment analysis, particularly emphasizing on the use of word embedding and machine learning algorithms for text classification.

In [10], the authors developed a multiclass text classification model using Naive Bayes classifier. The model is used to classify text in the Chinese language into different categories such as sport, tourism, automobile, etc. The model uses Vector Space Model to represent the text document as vectors. The result of the text classification model shows that it obtained the highest F1 score of 0.96 in the automobile category. The authors stated that the low F1 scores in other categories might be the result of some data which belong to multiple categories, and this issue might be addressed by the authors in future studies.

In [11], the authors were interested to compare how well do different word embedding architectures perform when utilized for text classification. The architectures used are Glove, Word2vec, and Fasttext. The reason for implementing these architectures in text classification is due to the fact that the generated word embeddings can capture semantic and contextual relationships between words. Random forest classifier is then trained using the word embeddings generated by each architecture is then tested to classify the tweets based on 4 emotion classes which are happy, surprised, angry, and sad. The study results show that the best performing model is the model with the parameters of 200 estimators in Random Forest classifier and a Fasttext word embedding with 300 dimensions. This model achieved the highest precision score of 91% and provides a satisfactory result for using word embeddings in text classification.

In [12], the authors were able to build word vectors based on sentiment lexicons for sentiment predictions in the Greek and English languages. Sentiment lexicons are used to give sentiment information when training the word vector model using Word2vec method. The proposed hybrid vectorization process, which combines lexicon-based features and word-embedding approaches, provides both the semantic and contextual relationships between words as well as the sentiment orientation of a document. The word vector model is then used in training the classifier model which uses Support Vector Machine (SVM) algorithm. The result shows that the proposed methodology does not surpass the state-of-the-art in the English language in terms of accuracy. It provides an accuracy improvement of 5.7% compared to the lexicon-based representations, and only 1.6% accuracy improvement compared to plain Word2vec representation. However, it gives high accuracy for the Greek language and is state-of-the-art in which it gives an accuracy improvement of 5.5% and 10.2% compared to lexicon-based representation and Word2vec representation respectively.

Similar to the previously mentioned study, in [13], the authors also used Word2vec to generate word embeddings and applied it in text classification using SVM classifier. This study aims to classify user interests based on the comments they posted on the social media platform Twitter. Comments are classified into 5 categories according to their topics which are sports, travel, fashion, food, and religion. The pre-labelled tweets used in this study are originally collected and published in [14]. After the word embeddings have been generated, it will then be inputted into Convolutional Neural Network architecture for deep features extraction. SVM classifier with linear kernel is then used to predict the sentiment of each tweet. The result of the proposed model shows that it achieved the highest accuracy of 97.3%.

After reviewing the existing works, the main steps for text classification from these works can be taken into consideration to be used in this study. The first main step would be to collect data that will be used in the study. The next step is to determine the word vectorization method and classifier. According to the results of the previous studies, using Word2vec and SVM classifier with linear kernel gives the best results when compared to other methods [15], and will therefore be used in this study.

3. METHODOLOGY

This study aims to build a classification model that is able to classify comment texts in Bahasa Indonesia into 5 different classes. As mentioned before, the 5 classes are based on a rating scale from 1-5. The following stages show the steps taken in this study to achieve its goal.

3.1 Data Collection

This research uses data collected using web scraping techniques. The data used are comments and ratings given by users in the review column found on the play.google.com web site. Using Python programming language, a web crawler is created in order to automatically scroll past multiple comments and to load more comments. The data collected includes:

- Title: It is the name of the application provider.
- User_name: the name of the user who made the comment.
- Rating: weight is given by the user after making a comment.
- Review_date: the time when the user makes a comment.
- Review_text: comments made by users.
- Reply_text: comment replies made by the admin.

After the data is collected, the data is stored in CSV format, after that the data is cleaned with the intention that the remaining data is only the data needed to become a model in training. The data divided into 2 files, namely training data and test data. Training data is used to create a model from word2vec. After the model is created the data is trained based on the ratings given by the user. The data can also be used to find out the sentiments of users who have made a comment. The results are compared and analysed to get the desired results.

3.2 Preprocessing

The next step after collecting data is to clean up the data by eliminate noise from the data so that only relevant information remains [16]. Preprocessing conducted in this research are case folding, tokenization, character cleansing, stemming, and stop word removal. Case folding is the process in which upper case letters will be transformed into their lowercase form. Since the data used are user-generated comments, there are numerous non-alphanumeric characters like emoticons that are used. Because of that, character cleansing must be done in order to remove these emoticons as well as punctuation and other special characters. The stemming process is also done in order to convert words into their root word form. The last preprocessing task is stop word removal, in which common words that are non-informative, such as conjunctions, will be removed. Most of the preprocessing tasks are done using the Natural Language Toolkit (NLTK) library. However, since the comment texts used are in the Indonesian language, a different library must also be used which is based on the Indonesian language. This research uses Sastrawi library for the stemming process and for eliminating stop words in the Indonesian language [17]. The difficulty in doing the preprocessing stage is eliminating various slang words in the Indonesian language, and so some slang words are added into the list of Indonesian stop words so they can be removed. Table 1 shows the examples of preprocessing tasks done in this research.

Table 1: An example of preprocessing comment

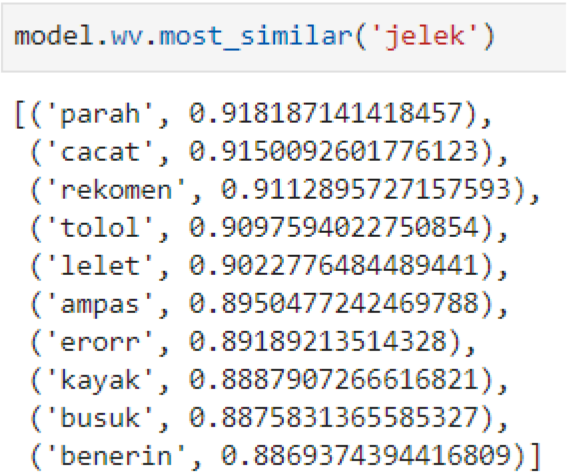
Original Comment	Harga naik, kualitas memburuk. Jaringan sering tidak stabil. Kecewa sih. :(
Comment Cleansing	harga naik kualitas memburuk jaringan sering tidak stabil kecewa sih
Stop Word Removal	harga kualitas buruk jaringan stabil kecewa
Stemming	harga kualitas buruk jaring stabil kecewa
Tokenizatio n	[harga], [kualitas], [buruk], [jaring], [stabil], [kecewa]

3.3 Word embedding using Word2vec

At this stage, the data which has been preprocessed is used in the word embedding process. To implement Word2vec for the word embedding process, the Gensim library is used. This process takes the preprocessed data in the form of tokenized texts as input and produces a model as output. This model is in the form of a vocabulary of words with vectors attached to each of the words. The number of vector dimensions depends on the parameters set when training the model. Parameters used for training the Word2vec model can be seen in Table 2. By using the “most_similar” function from Gensim, a target word can be inputted, and the function returns 10 closest words from the target word. For testing the model, we used the word “jelek”, which means “poor quality”. The model shows promising results in which the “most_similar” function returns 10 words that have a relatively similar meaning to the word “jelek”. The result of this testing can be seen in Figure 1.

Table 2: Parameters used in training Word2vec model

Parameter	Description	Value
size	Number of vector dimensions	100
window	Context word window size	2
workers	Number of parallel threads	4
min_count	Minimum word count	1
iter	Number of training iterations	40



```
model.wv.most_similar('jelek')

[('parah', 0.918187141418457),
 ('cacat', 0.9150092601776123),
 ('rekomen', 0.9112895727157593),
 ('tolol', 0.9097594022750854),
 ('lelet', 0.9022776484489441),
 ('ampas', 0.8950477242469788),
 ('erorr', 0.89189213514328),
 ('kayak', 0.8887907266616821),
 ('busuk', 0.8875831365585327),
 ('benerin', 0.8869374394416809)]
```

Figure 1: Word2vec most similar result

3.4 SVM Algorithm

After the word embedding has been obtained, it is then used in training the classification model. One of the machine learning algorithms, SVM algorithm (Linear SVC), is used for classifying the comment texts. The word embedding that has been obtained through the use of Word2vec, will be fitted in the classification model by the use of a vectorizer [18]. The classifier will transform features obtained from the word embedding process into a high dimensional feature space [19]. The obtained comment data from play.google.com website is split into training data and testing data with a ratio of 80:20 respectively. The model is trained using comment texts that have their class labels. The classes are in the form of a rating scale from Class 1-5 which is based on the Google Play Store rating. After training the classification model, it is then tested using the test data using the prediction function from the Scikit-learn library. The testing process gives a result in the form of the accuracy, precision, recall, and F1 score of the classification model.

4. RESULT ANALYSIS

4.1 Datasets

The total amount of data collected from play.google.com is 16,360 comments along with each of their ratings. The data count for each rating is shown in Table 3.

Table 3: Data count for each rating

Rating	Data Count
1	2765
2	546
3	924
4	2498
5	9627

The word vector model has 100 dimensions and is trained using Word2vec with 40 iterations. The vector representations for each word were successfully obtained, and it shows promising results based on some tests done using functions from Gensim library such as “most_similar” and “distance”.

The data obtained was split into training data and test data with a ratio of 80:20 respectively, in which the amount of training data is 13,088 and the amount of test data is 3,272. In this paper, 4 different experiments were carried out: training data with 5 classes and using stop word removal; training data with 5 classes without using stop word removal; training data with 3 classes, using stop word removal, and undersampling data; training data with 3 classes and using stop word removal.

4.2 Experiment Results

For the first experiment, the model was trained to classify the data into 5 classes. Afterwards, the model was tested using the test dataset and achieved an F1 score of 0.549. Table 4 shows the confusion matrix for the first experiment.

Table 4: First confusion matrix

		Predicted				
		1	2	3	4	5
Actual	1	260	0	1	0	273
	2	50	0	0	0	66
	3	37	0	1	0	160
	4	40	0	2	0	457
	5	59	0	4	0	1862

The result of the first experiment was quite unsatisfactory. As in [20], the second experiment was conducted without eliminating stop words to try improving the result. The second experiment resulted in an F1 score of 0.556, and Table 5 shows its confusion matrix.

Table 5: Second confusion matrix

		Predicted				
		1	2	3	4	5
Actual	1	249	100	0	21	188
	2	43	9	0	9	42
	3	45	20	0	8	132
	4	39	18	0	29	418
	5	58	39	0	80	1725

The second experiment which was conducted without stop word elimination did not have a significant difference in the result compared to the first experiment. Since the data count for each of the class is unbalanced, the third experiment conducted will also implement undersampling. Imbalance data is a condition where a class has a size that is much larger than other classes. Undersampling reduces the size of large classes, while oversampling increases the size of small classes [21]. Besides implementing undersampling techniques, the classes have also been merged to only form 3 classes. Classes 1 and 2 are merged to form the new Class 1, while Classes 4 and 5 are merged to form the new Class 5. The result of the third experiment is an F1 score of 0.539 and Table 6 shows its data and Table 7 shows its confusion matrix.

Table 6: Data count for new classes after undersampling

Class	Data Count
1	924
2	924
3	924

Table 7: Third confusion matrix

		Predicted		
		1	**2**	**3**
Actual	**1**	115	42	33
	2	44	54	91
	3	12	31	133

The result of the third experiment is still not significantly different than previous experiments. In fact, the F1 score after implementing undersampling techniques declined. For the fourth experiment, the classes will still be merged, and the model will only be trained to classify 3 classes just like in the previous experiment. However, undersampling techniques will not be used in the fourth experiment. The data count for the new classes are shown in Table 8.

Table 8: Data count for merged classes

Class	Data Count
1	3311
2	924
3	12125

The fourth experiment shows a significant increase in the performance of the model. The F1 score of the fourth experiment is 0.795 and this experiment obtained the highest precision, recall, and F1 score compared to the previous experiments. Table 9 shows the confusion matrix for the fourth experiment.

Table 9: Fourth confusion matrix

		Predicted		
		1	**2**	**3**
Actual	**1**	415	9	252
	2	49	2	119
	3	155	14	2257

5. CONCLUSION

In this paper, 4 different experiments were carried out: training data with 5 classes and using stop word removal; training data with 5 classes without using stop word removal; training data with 3 classes, using stop word removal, and undersampling data; training data with 3 classes and using stop word removal and the best F1 score result is 0.795. With the result of this study, it is expected that the model can be used on social media that do not require users to give a score on the given review. The model is expected to be able to automatically classify comments according to their predicted ratings. So in the future, it is expected that sentiment analysis can be done even with comments that do not include the ratings given by users, and it is in the hope that it will ease the process of sentiment analysis for businesses in order to evaluate their products and services.

For future works, different word embedding models such as Glove or Fasttext can be used for extracting word embeddings. Deep learning methods as well as other machine learning algorithm for classifying text can also be used. Since many of the words from the comment texts are not in standard form, building a dictionary for slang words in Bahasa Indonesia is also recommended in order to help improve classifier accuracy.

REFERENCES

1. F. S. Khoo, P. L. The, and P. B. Ooi. **Consistency of online consumers' perceptions of posted comments: An analysis of tripadvisor reviews**, *Journal of ICT*, pp. 374–393, 2017.
2. S. Kim, and H. Park. **Effects of various characteristics of social commerce (s-commerce) on consumers' trust and trust performance**, *International Journal of Information Management*, 33, pp. 318– 332, 2013.

3. Y. Zhong. **Social commerce: A new electronic commerce**, *Eleventh Wuhan International Conference on e-Business*, 49, pp. 164-169, May 2012.
4. D. Sehrawat, and N. S. Gill. **Review and comparative analysis of topic identification techniques**, *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 8, pp. 995-803, 2019. https://doi.org/10.30534/ijatcse/2019/71832019
5. F. Taheri and M. A. Shourmasti. **Effects of various characteristics of social commerce on consumers' trust and trust performance**, *International Academic Journal of Business Management*, vol. 3, no. 3, pp. 20-26, 2016.
6. A. S. Al-Adwan and M. A. Al-Horani. **Boosting customer e-loyalty: An extended scale of online service quality**, *Information*, vol. 10, no. 12, 2019.
7. C. Grosso and C. Forza. **Users' social-interaction needs while shopping via online sales configurator**, *International Journal of Industrial Engineering and Management (IJIEM)*, vol. 10, no 2, pp. 139-154, June 2019.
8. G. Aydin. **Examining social commerce intentions through the uses and gratifications theory**, *International Journal of e-Business Research*, vol. 15, pp. 44-70, 2019.
9. M. J. M. Razi, M. Sarabdeen, M. I. M. Tamrin, and A. C. M. Kijas. **Influencing factors of social commerce behavior in saudi arabia**. *2019 International Conference on Computer and Information Sciences (ICCIS)*, pp. 1-4, 2019.
10. G. Zheng and Y. Tian. **Chinese web text classification system model based on naive bayes**, *2010 International Conference on E-Product E-Service and E-Entertainment (ICEEE)*, pp. 1–4, 2010.
11. P. Vora, M. Khara, and K. Kelkar. **Classification of tweets based on emotions using word embedding and random forest classifiers**, *International Journal of Computer Applications*, vol. 178, no. 3, pp. 1-7, 2017.
12. M. Giatsoglou, M. G. Vozalis, K. Diamantaras, A. Vakali, G. Sarigiannidis, and K. C. Chatzisavvas. **Sentiment analysis leveraging emotions and word embeddings**, *Expert Systems with Applications*, vol. 69, pp. 214–224, 2017.
13. A. H. Ombabi, O. Lazzez, W. Ouarda, and A. M. Alimi. **Deep learning framework based on word2vec and cnn for users interest classification**, *2017 Sudan Conference on Computer Science and Information Technology (SCCSIT)*, pp. 1-7, 2017.
14. B. Pang and L. Lee. **A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts**, *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, pp. 271–278, 2004.
15. M. Islam, and N. Sultana. **Comparative study on machine learning algorithms for sentiment classification**, *International Journal of Computer Applications*, vol. 182, no. 21, pp. 1-7, Oct. 2018.
16. P. Thakur, and R. Shrivastava. **A review on text based emotion recognition system**, *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 7, pp. 67-71, 2018. https://doi.org/10.30534/ijatcse/2018/01752018
17. E. Setiani, and W. Ce. **Text classification services using naïve bayes for Bahasa Indonesia**, *2018 International Conference on Information Management and Technology (ICIMTech)*, pp. 361-366, 2018.
18. J. Lilleberg, Y. Zhu, and Y. Zhang. **Support vector machines and word2vec for text classification with semantic features**, *2015 IEEE 14th International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC)*, pp. 136-140, 2015. https://doi.org/10.1109/ICCI-CC.2015.7259377
19. I. Ferwana. **Clustering Arabic Tweets for Saudi national vision 2030**, *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 8, pp. 243-248, 2019. https://doi.org/10.30534/ijatcse/2019/3781.42019
20. I. N. Habibi, and A. S. Girsang. **Classification of tourist comment using word2vec and random forest algorithm**, *Journal of Environmental Management and Tourism*, vol. 9, pp. 1725-1732, 2018.
21. A. Agrawal, H. Viktor, and E. Paquet. **SCUT: Multi-class imbalanced data classification using SMOTE and cluster-based undersampling**, *2015 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K)*, pp. 226-234, 2015. https://doi.org/10.5220/0005595502260234