IJCCS (Indonesian Journal of Computing and Cybernetics Systems)
Vol.13, No.1, January 2019, pp. 43~52
ISSN (print): 1978-1520, ISSN (online): 2460-7258
DOI: https://doi.org/10.22146/ijccs.40125

Hate Speech Detection for Indonesia Tweets Using Word Embedding And Gated Recurrent Unit

Junanda Patihullah*[1], Edi Winarko[2]
[1]Program Studi S2 Ilmu Komputer FMIPA UGM, Yogyakarta, Indonesia
[2]Departemen Ilmu Komputer dan Elektronika, FMIPA UGM, Yogyakarta, Indonesia
e-mail: ***[1]jpatihullah@gmail.com**, [2]ewinarko@ugm.ac.id

Abstrak

Media sosial telah mengubah cara orang dalam mengekspresikan pemikiran dan suasana hati. Seiring meningkatnya aktifitas pengguna sosial media, tidak menutup kemungkinan tindak kejahatan penyebaran ujaran kebencian dapat menyebar secara cepat dan meluas. Sehingga tidak memungkinkan untuk mendeteksi ujaran kebencian secara manual. Metode Gated Recurrent Unit (GRU) adalah salah satu metode deep learning yang memiliki kemampuan mempelajari hubungan informasi dari waktu sebelumnya dengan waktu sekarang. Pada penelitian ini fitur ekstraksi yang digunakan adalah word2vec, karena memiliki kemampuan mempelajari semantik antar kata. Pada penelitian ini kinerja metode GRU dibandingkan dengan metode supervised lainnya seperti Support Vector Machine, Naive Bayes, Random Forest dan Regresi logistik. Hasil yang didapat menunjukkan akurasi terbaik dari GRU dengan fitur word2vec adalah sebesar 92,96%. Penggunaan word2vec pada metode pembanding memberikan hasil akurasi yang lebih rendah dibandingkan dengan penggunaan fitur TF dan TF-IDF.

***Kata kunci**—Gated Recurrent Unit, hate speech, word2vec, RNN, Word Embedding*

Abstract

Social media has changed the people mindset to express thoughts and moods. As the activity of social media users increases, it does not rule out the possibility of crimes of spreading hate speech can spread quickly and widely. So that it is not possible to detect hate speech manually. GRU is one of the deep learning methods that has the ability to learn information relations from the previous time to the present time. In this research feature extraction used is word2vec, because it has the ability to learn semantics between words. In this research the Gated Recurrent Unit (GRU) performance is compared with other supervised methods such as Support Vector Machine, Naive Bayes, Random Forest, and Logistic Regression. The result of experiments shows that the the combination of word2vec and GRU gives the best accuracy of 92.96%. However, the used of word2vec in the comparison methods results in the lower accuracy than the used of TF and TF-IDF features.

***Keywords**—Gated Recurrent Unit, hate speech, word2vec, RNN, Word Embedding*

Received October 26th,2018; Revised January 23th, 2019; Accepted January 29th, 2019

1. INTRODUCTION

Social media as a means of communication can disseminate information quickly and widely, making it not only as a means of friendship and various information, but also used as a means of trading, dissemination of government policies, political campaigns and religious preaching [1]. With the increasing activity of social media users, it does not rule out the possibility of cyber crime such as the dissemination of information containing hate speech. Hate speech on the social media can be in the form of words that contain hatred in writing and shown to individuals or groups to the detriment of the targeted party. Detecting hate speech is very important to analyze public sentiments from certain groups towards other groups, so as to prevent and minimize unwanted actions or things [2].

Detection of hate speech for Indonesian language has been done before, using bag of words, namely word n-gram and character n-gram. Machine learning algorithms used for classification are, Bayesian Logistic Regresion, Naive Bayes, Support Vector Machine and Random Forest Decision Tree. Currently, the highest F-measure was achieved when using word n-gram, especially when combined with Random Forest Decission Tree (93.5%), Bayesian Logistic Regresion (91.5%) and Naive Bayes (90.2%) [3]. Detection of hate speech of Indonesian language can also be done using backpropagation neural network algorithm with a combination of lexicon based and bag of words features with the highest accuracy obtained at 78.81%. [4]. In this paper, we propose the combination of word embedding as our feature and Gated Recurrent Unit (GRU) as our classifier for hate speech detection in Indonesian Tweets.

2. METHODS

In this section, we discuss architecture and methods used to detect hate speech in Indonesia tweets. The main stages in this research are three parts, preprocessing, feature extraction and classification, as can be seen in Figure 1. Each of this part is described in the following subsection.

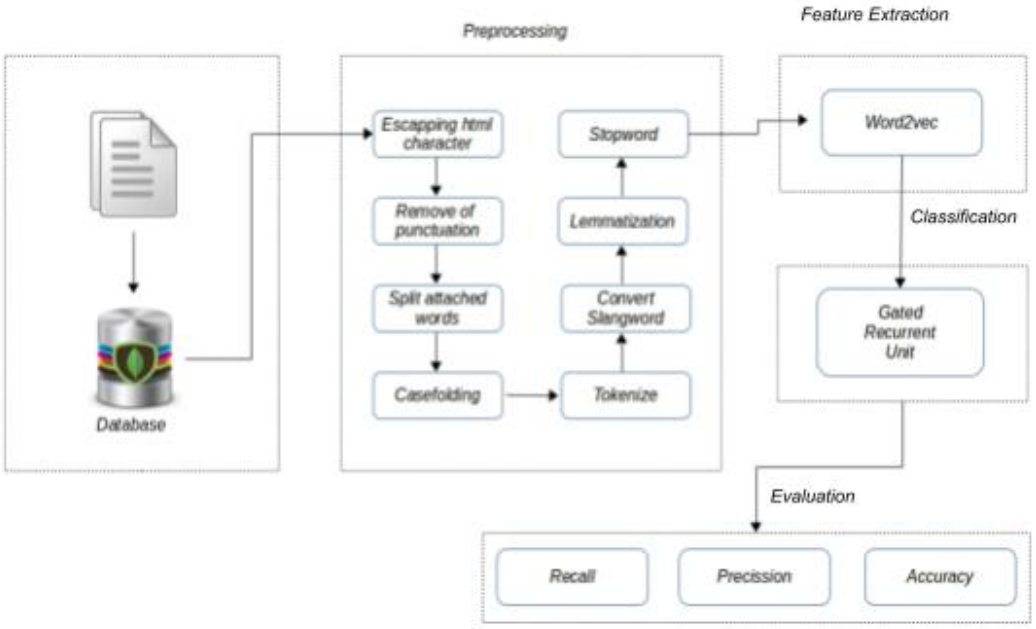


Figure 1. Hate Speech Detection Arsitecture

2.1 Preprocessing

Preprocessing stage is very important in classification to get the best model. The tweet processing consists of several steps: 1) Escaping html characters; 2) Removal of punctuation; 3) Split attached words; 4) Case folding; 5) Tokenization; 6) Convert slangwords; 7) Removal of stop-words. Escaping html character aims to remove URL link and also html character that often found in tweets. Remove of punctuation is used to delete special characters that are often found in tweets such as hastag (#), @user, retweet (RT). Beside that at this stage will also removing punctuation. Split attached words,we humans in the social forums generate text data, which is completely informal in nature. Most of the tweets are accompanied with multiple attached words like RainyDay, PlayingInTheCold etc. These entities can be split into their normal forms using simple rules and regex. Case folding is a proses of converting all characters into lowercase. Tokenize is a task of splitting text into smaller units. Convert slangwords, is a process to be transformed of a majority slang words into standard words. The next step is stopword removal. Stopwords is words on uninformative, these words will be remove based on the existing stoplist dictionary. This research is using the stop-word list from Rahmawan [5].

2.2 Feature Extraction

Word2Vec is the name of the word vector representation defined by Mikolov *et al.* [8]. The main basis or component for generating vector values in word2vec is artificial neural networks built from CBOW and Skip-gram architectures. Before word2vec can represent the vector value for each word, word2vec will first create a model of the word distribution during training using Indonesian documents collected from Wikipedia. The number of documents used is 1,120,973. In order to build the word2vec feature model, there are three processes involved, i.e., vocabulary builder, context builder, and neural network. Figure 2 shows the three processes in the word2vec model building.

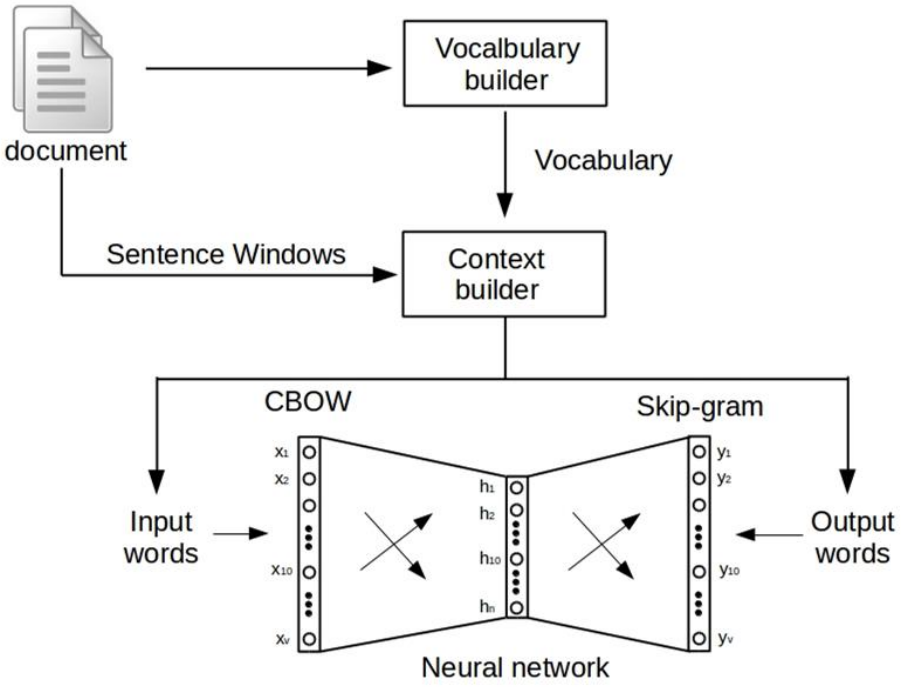


Figure 2. word2vec's main architecture

2.2.1 Vocabulary builder

The vocabulary builder is the first building block of the word2vec model. It takes raw text data, mostly in the form of sentence. The vocabulary builder is used to build vocabulary from text corpus. It will collect all the unique words from corpus and build the vocabulary. In this vocabulary builder section, the data used is document that has been downloaded from the Wikipedia. The result of the vocabulary builder process is a dictionary of words with a word index and the occurrence value of each word [7].

2.2.2 Context builder

The context builder uses output of the vocabulary builder. Contenxt builder is a process to find out the relationship between the appearance of one word with other words around it by using the concept of the context window or also called a sliding window. In general, the size of context windows in NLP is 5 to 8 neighboring words. If we choose the size of the window content is 5, then 5 words that appear on the left and right of the center word. In this research, the size of the content window used is 5. Table 1 gives an example of the content window with the window size of 1. The underlined word is the center word. The results of the content window of the content builder will be used in the next process, namely the neural network section.

Table 1 Examples context window

Text	Word pairing
I like deep learning.	(I, like)
I *like* deep learning.	(like, deep), (like, I)
I like *deep* learning.	(deep, learning), (deep, like)
I like deep *learning*.	(learning, .), (learning, deep)

Furthermore, the results of the content window of the content builder will be used in the next process, namely the neural network section.

2.2.3 Neural networks (CBOW and Skip-Gram architecture)

Word2vec uses an artificial neural network architecture formed from CBOW and Skip-gram architectures. This artificial neural network is used to conduct training so that each word can be represented by a vector. In this case the neural network architecture uses 3 layers, input layer, hidden layer and ouput layer [8]. In this research, the hidden layer contains 200 neurons and the output layer has the same amount as the input layer. Input for the network is the value of each word that has been converted into one-hot encoding. Figure 3 shows the neural network architecture to generate word2vec.

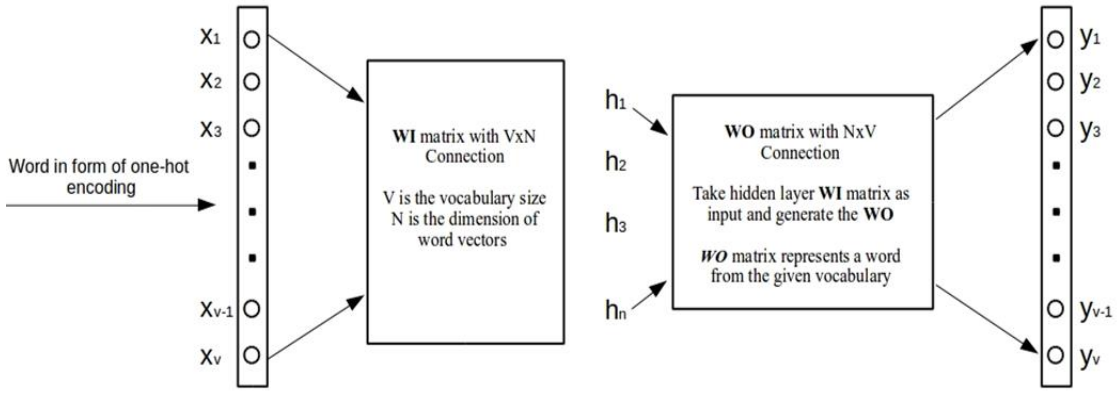
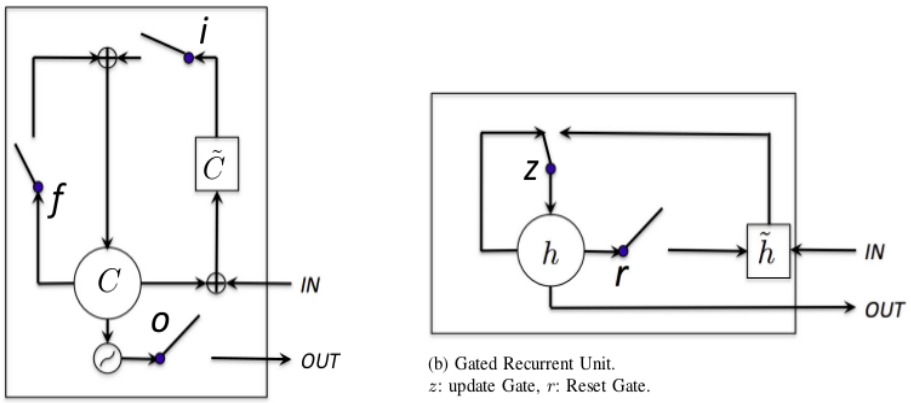


Figure 3. Neural network input and output structures to create word2vec

2.3 Classification

This research used GRU to detect hate speech in the Indonesian language. GRU is a variation on the LSTM that is simpler than LSTM, and in some cases produce equally excellent results. As LSTM, GRU (Gated Recurrent Unit) aims to solve the vanishing gradient problem which comes with a standard recurrent neural network. GRU combines the forget gate and input gate into one update gate and has an additional reset gate as shown in Figure 4. The GRU is increasingly popular and many use it to solve NLP problems [9].



(a) Long Short Term Memory Unit. i: Input Gate, f: Forget Gate, o: Output Gate.

(b) Gated Recurrent Unit. z: update Gate, r: Reset Gate.

Figure 4. Gated Recurrent Unit architecture

To solve the vanishing gradient problems of a standart RNN, GRU uses update gate and reset gates. Basically, these are two vectors which decide what information should be passed to the output. The activation function $h_t^j$of the GRU at time t is a linear interpolation between previous activation function $h_{t-1}^j$and candidate output activation $\widetilde{h_t^j}$as seen in Equation (1)

$$h_t^j = z_t^j \circ h_{t-1}^j + (1 - z_t^j) \circ \widetilde{h_t^j} \quad (1)$$

The function update gate $z_t^j$is to decide how many previous units must be kept, as can be seen in Equation (2).

$$z_t^j = \sigma(W_z x_t + U_z h_{t-1})^j \quad (2)$$

When $x_t$is plugged into the network unit, it is multiplied by its own weight W_z. The same goes for h_{t-1}which hold the information for the previous t-1 units and is multiplied by its own weight U_z.

The function reset gate r_t^j is used from the model to decide how much of the past information to forget, as can be seen in Equation (3). This function is the same as the one for the update gate z_t^j. The difference comes in the weights W_r, $U_r$and the gates usage.

$$r_t^j = \sigma(W_r x_t + U_r h_{t-1})^j \quad (3)$$

Activation function of candidates ouput $\widetilde{h_t^j}$calculates the value of the unit before it is decided to be updated or not and (∘) shows the Hadamard product multiplication element. Function activation candidate ouput can seen in Equation (4).

$$\widetilde{h_t^j} = \tanh(W x_t + r_t \circ U h_{t-1})^j \quad (4)$$

3. RESULTS AND DISCUSSION

This research used Twitter hate speech in Indonesian language tthat have been collected and labelled by [3]. The number of tweet data is 713 data, 260 tweets is labeled as hate speech 453 is labeled as non hate speech tweets.

A. Comparison Word2vec with TF and TF-IDF

In the first experiment we will try to compare the word2vec feature with TF and TF-IDF to find out the ability of word2vec as a feature in the classification model. Supervised algorithms that will be used for this experiment Support vector machines, Naive Bayes, Bayesian Logistic Regression and Random Forest. This experiment is carried out based on the assumption that word2vec has a better ability to detect hate speech compared to other features, namely, TF and TF-IDF.

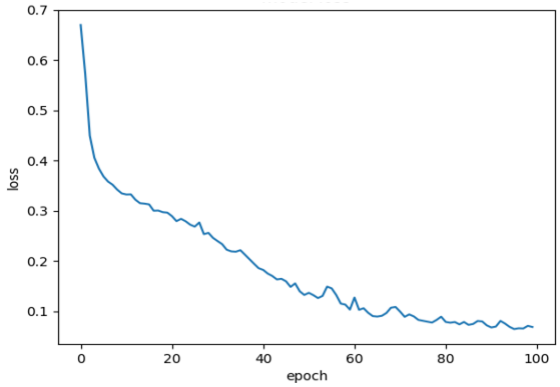
Table 2 Comparison of word2vec against TF and TF-IDF

Feature	Accuracy %			
	SVM	NB	BLR	RFDT
Word2vec	73.07	77.88	73.07	79.80
TF	83.65	79.80	78.84	81.73
TF-IDF	80.76	78.80	80.76	82.69

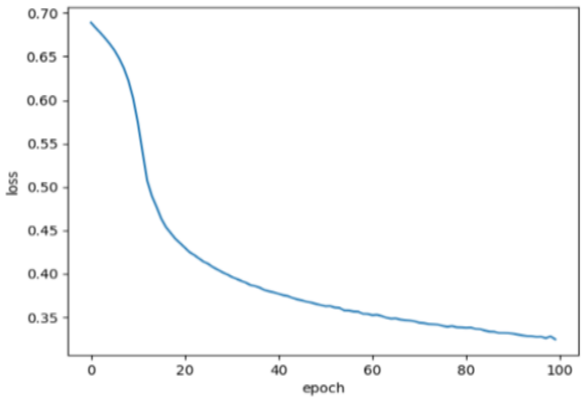
Experiment results can seen in Table 2. This table shows that the highest accuracy of the word2vec feature is achieved by using the random forest algorithm, with an accuracy value of 79.80%. This accuracy is lower than the accuracy of using TF and TF-IDF on all algorithms. This baseline experiments show that for classical algorithms used in this research, word2vec feature result in lower accuracy compared to TF and TF-IDF features.

B. Determining Learning Rate

Experiments to determine the learning rate were performed with single GRU layer by setting number of neuron 200, and epoch 100. The amount of the learning rate is certainly not too big and not too small. The choice of a large learning rate will make the learning process not too optimal, while the learning rate value that is too small can cause the training process to be less good in time complexity. The leaning rate value is set to 0.001, 0.0001, and 0.00001.



a. Learning rate 0.001



b. Learning rate 0.0001

Figure 5. The effect of learning rate on loss value

The results of the experiment using learning rate 0.001 and 0.0001 are shown in Figure 5. Model with learning rate value of 0.00001 is not able to achieve the expected convergent training loss value. The reduction of the learning rate value further makes the convergent model and loss value closer to zero but the time needed for training is longer. Therefore in the next experiment the learning rate chosen is 0.001.

B. Determining number of neuron in hidden layer

This experiments is used to determine the optimal number of neuron in hidden layer by setting the learning rate to 0.001. We use GRU with 1 and 2 architectures. The number of neurons in hidden layers to be tested is 128, 200, 250 and 300.

The result of our experiment can seen Figure 6. The result shows that the addition or reduction of the number of neurons can affect the accuracy of the model. The initial accuracy GRU with 1 layer is 90.28% and increases with the addition of the number of neurons. In contrast, the accuracy obtained by GRU with 2 layer is the highest with the value of 92.96 when the number of neurons is 200. The addition of the number of neurons can not increase the accuracy.

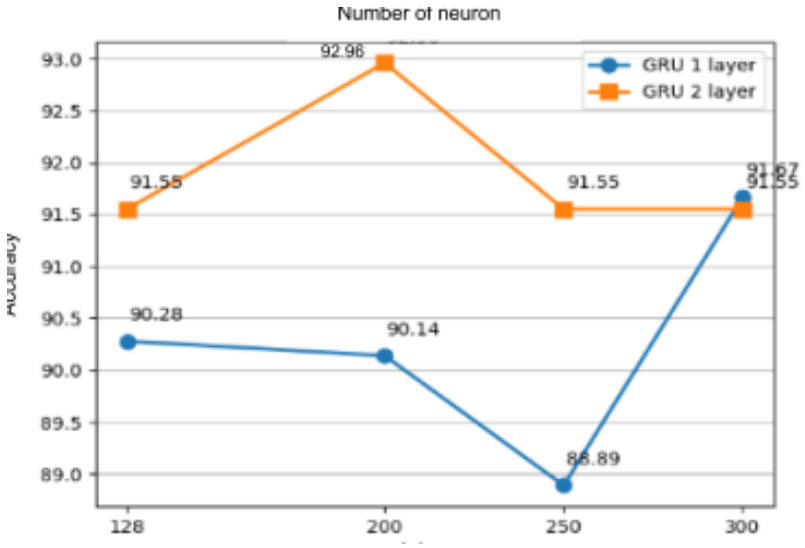


Figure 6. Effect of number of hidden layer neurons on accuracy

C. Overall model performance

Table 4 shows the best performance of the GRU and classical algorithms used in our experiment. All of the values in this table is the average values from the experiment using 10-fold cross validation on 713 training data. The best performance of GRU model is achieved by GRU with 2 layer, with the learning rate of 0.001, 200 neurons in the hidden layer, which has the accuracy of 92.96%. This shows that the ability of the GRU model is better because the GRU model is built by having an update gate and a reset gate that can store and dispose of previous data. Function that is owned by the update gate and reset gate makes the GRU model can know the information in the previous time and the current time information so that it can increase accuracy in determining the class on the tweet.

Table 4. esult compared models

	Precision (%)	**Recall (%)**	**F-measure (%)**	**Accuracy (%)**
GRU	88.46	92.00	90.20	92,96
SVM	73.08	90.48	80.85	83.65
NB	76.92	86.96	81.63	79.80
BLR	69.23	100.00	81.82	78.84
RFDT	84.62	80.00	86.27	82.69

4. CONCLUSION

Based on experiment result from this study, it can be conclude that Gated Recurrent Unit with word2vec feature better as compared to traditional supervised learning for detect hate speech in Indonesian Language. Feature extraction using word2vec is able to produce semantic values for each word in other words that have the same meaning so that the classification results obtained are quite good. The lack of word2vec is found in word2vec's dependence on training data, the more training data, the greater the chance for word2vec to be able to represent all the desired words.

REFERENCES

[1] G. A. Buntoro, Analisis Sentimen Hate Speech Pada Twitter Dengan Metode Naive Bayes Classifier dan Support Vector Machine, Jurnal Dinamika Informatika, volume 5, no.2, 2016 [Online]. Available: http://ojs.upy.ac.id/ojs/index.php/dinf/article/viewFile/975/775.

[2] P. Badjati, S. Gupta, M. Gupta and V. Varma, Deep Learning for hate Speech Detection in Tweets, Proceedings of the 26th International Conference on World Wide Web Companion, pp. 759-760, doi: 10.1145/3041021.3054223, 2017.

[3] I. Alfina, R. Mulia, M. I. Fanany and Y. Ekanata, Hate Speech Detection in the Indonesian Language: A Dataset and Preliminary Study, 9th Int. Conf. Adv. Comput. Sci. Inf. Syst. (ICACSIS 2017), 2017 [Online]. Available: https://ieeexplore.ieee.org/document/8355039.

[4] M. M. Munir, M. A. Fauzi and R. S. Perdana, Implementasi Metode Backpropagation Neural Network berbasis Lexicon Based Features dan Bag Of Words Untuk Identifikasi Ujaran Kebencian pada Twitter, Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, volume 2, no.10, pp. 3182-3191, 2018 [Online]. Available: http://j-ptiik.ub.ac.id/index.php/ j-ptiik/article/view/2573.

[5] A. D. Rahmawan, Analisis Emosi Pada Tweet Berbahasa Indonesia Tentang Ulasan Film, Tesis, Program Studi S2 Ilmu Komputer, Universitas Gadjah Mada, Yogyakarta, 2018.

[6] M. Seok, H. Song, C. Park, J. Kim and Y. Kim, Named Entity Recognition Using Word Embeddings as a Feature, International Journal of Software Engineering and Its Application (IJSEIA), volume 10, no.2, pp. 93-104, 2016 [Online]. Available: http://www.sersc.org/journals/IJSEIA/vol10_no2_2016/8.pdf.

[7] Thanaki, Python Natural Language Processing, Packt Publishing, 2017.

[8] T. Mikolov, K. Chen, G. Carrado and J. Dean, Efficient estimation of word representations in vector space , arXiv preprint arXiv:1301.3781v3 [cs.CL], 2013 [Online]. Available: https://arxiv.org/abs/1301.3781 .

[9] R. Rana, J. Epps, R. Jurdak, X. Li, R. Geocke, M. Brereton and J. Soar, Gated Recurrent Unit (GRU) for Emotion Classification from Noisy Speech, arXiv preprint arXiv:1612.07778v1 [cs.HC], pp. 1-9, 2016 [Online]. Available: https://arxiv.org/abs/1612.07778 .

[10] J. lilleberg, Y. Zhu, Y. Zhang, Support Vector Machines and Word2vec for Text Classification with Semantic Feature, Proc.2015 IEEE 14th International Conference on Cognitive Informatics and Cognitive Computing [ICCI'CC'15], 2015 [Online]. Available: https://ieeexplore.ieee.org/document/7259377.