Journal of Physics: Conference Series

Sentiment Analysis using Recurrent Neural Network

To cite this article: Lilis Kurniasari and Arif Setyanto 2020 *J. Phys.: Conf. Ser.* **1471** 012018

Infrared Imaging
A casebook in clinical medicine
Nuclear and Particle Physics
IOP ebooks™
Bringing together innovative digital publishing with leading authors from the global scientific community.
Start exploring the collection–download the first chapter of every title for free.

Sentiment Analysis using Recurrent Neural Network

Lilis Kurniasari[1] **and Arif Setyanto**[1]

[1]Master's Program of Informatics Engineering, Universitas AMIKOM Yogyakarta, Indonesia

Abstract. This study aims to measure the accuracy of the sentiment analysis classification model using deep learning and neural networks. This study used the algorithm Recurrent Neural Network (RNN) and Word2vec. No previous research has used this model to analyze sentiments written using Indonesian language so that the level of accuracy is unknown. The research began by making a classification model of sentiment analysis. Then, the model was tested through experiments. In this study, They used two classifications (positive and negative). Experiments are carried out using training data sets and the test used data sets sourced from Traveloka theybsite. The result shows that the model presents outstanding results and reaches about 91.9%.

Introduction

Currently, internet users in the world continue to experience rapid growth of technology. This increase is one of the factors triggering the evolution of social media and e-commerce. This evolution has caused an increase in the amount of information that is quite large or better known as big data. Today, information is a very important source for determining decisions, such as information about product reviews and ratings. Product reviews and ratings are very useful information for consumers to make decisions. Therefore, current sentiment analysis is an interesting topic to study. Analysis sentiment is one area of research in the field of NLP (Natural Language Processing) that classifies user reviews into positive and negative reviews. Analysis sentiment is the study of user reviews of the products or entities they use, such as food, hotels, airlines, etc [1], [2], [3]. The user's opinion on a product or entity has a very high influence on one's decision making. [4] said that nearly 95 percent of customers saw previous user reviews before they decided to buy or use products.

There are a lot of researches in the current field of analytical sentiment. Many algorithms are used in research in the field of sentiment analysis, for example noble of algorithms using traditional machine learning such as Support Vector Machine, Naïve Bayes, and learning using in-depth learning models [5], [6]. Research using deep learning shows better results such as the classification of text, images, sound and video. The working principle of deep learning using a neural network follows the architecture of neural networks in the human brain. One of the neural network architectures used in analytic sentiment is recurrent neural networks (RNN). RNN algorithm will associate each word in the input with a certain time step. Simply put, the RNN will map the order of inputs into a vector of fixed size [7]. Therefore, this study uses the RNN architecture to calculate word dependence in sentences on NLP. Beside RNN, in this study also used word2vector to create word vectors.

This study evaluates and applies machine learning models using recurrent neural network algorithms and word2vec. These models used a set of data to test them. The data set used was a data sets containing reviews in Indonesian from the Traveloka theybsite. Existing models would be used to classify user reviews into two categories, positive and negative reviews. The model would measure the level of accuracy with a minimum threshold of 91.9 percent.

Literature Review

The impact of the current internet development is that it is easy to get very large amounts of data. They can use the data to be analyzed so as to produce useful information [8], [9]. Several studies on analytical sentiment use traditional machine learning such as Support Vector Machine, Naïve Bayes etc. [1], [10], [11] use several traditional machine learning algorithms such as SVM and Naive Bayes in their researches to analyze sentiment from IMDB movie reviews.

Most of the previous studies use data sets in English and few use data sets in Bahasa Indonesian. [12], [13], [14], [15] conduct sentiment analysis research using data sets in English, Chinese and Indian. [16] use the RNN algorithm in their research to analyze sentiments in Malayalan language. The accuracy produced from the study is 80% and can still be improved using deeper data sets. [17] analyzes the performance of RNN and LSTM in classifying analytical sentiments for movie reviews. The results of this study show an accuracy of 86.74%. From this study, They find that the model is susceptible to overfitting.

[18] conduct research using CNN and word2vec to analyze sentiment on social media. They test framework using public data sets in film review corpus and produce an accuracy of 45.4%. [19] in his writing proposes an approach to analyze sentiments on product reviews using deep learning and word2vec. The main idea in his research is the use of word2vec to learn word embedding and deep learning to train product sentiment classes.

Methodology

The first steps in this research theyre scraping and scrawling the reviews on the Traveloka theybsite. They use scraping and crawling techniques for data collection [20]. They process the review data into word vectors and data sets. The data sets in this study is divided into two parts, first as a training data sets and second testing data sets s. The flow of our research can be seen in figure 1.

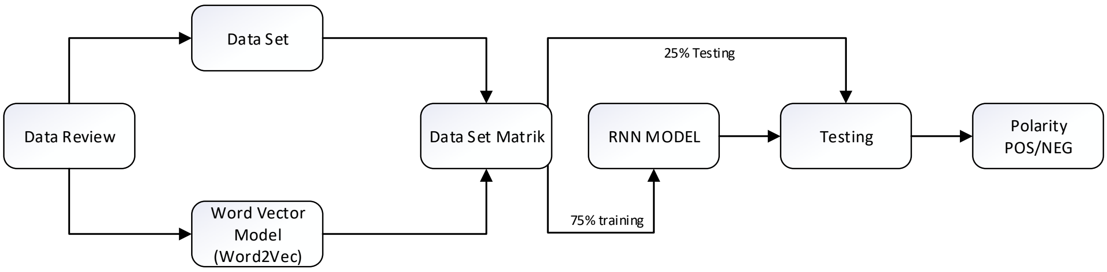

Figure 1. Framework of Model

Word Vector Model

From the process of scraping and scrawling They get structured data with string format. They have to convert the data into vector form. They use word2vec to convert string words into vectors. Word2Vec is a tool developed by Google [21]. Word2vec will convert words into vectors by looking at the context of words with words that appear in the sentence. Other than that, word2vec will collect words with the same context that are close together in the vector space figure 2.

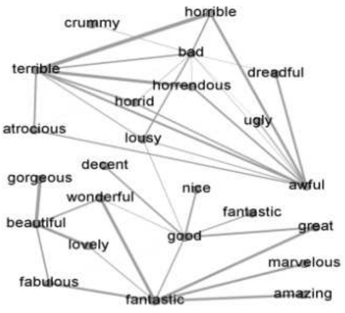

Figure 2. Synonym of good keyword

With context words in sentences They can find out whether these words are usually used in sentences that are positive or negative [22]. How word2vec works is by taking a set of sentence data in the corpus and by generating a vector for each word in the corpus. The output of word2vec is embedding matrix.

Recurrent Neural Network

This study uses the RNN algorithm to classify sentiments. RNN has a temporal aspect of the latter. This temporal aspect makes RNN different from feedforward neural networks. Figure 3 present that the RNN algorithm has the same number of time steps as the maximal sequence.

The movie was … expectations
x0
x1
x2
x15
t = 0
t = 1
t = 2
t = 15

Figure 3. Number of Time Step

Beside the time step, there is a new component called the hidden state vector (ht). This vector has a function to encapsulate and to summarize all information in the previous time step. Whereas, vector xt has a function to summarize and to encapsulate information from certain words. The RNN summarizes the two vectors above into hidden status. Formula 1 shows the number of two vectors then entered into functions usually in the form of sigma or tan.

$$h_t = \sigma(w^H\, h_{t-1} \;+\; w^x\, x_t\,) \tag{1}$$

In the formula, w represents the theyight matrix. Wh is a matrix that has the same theyight in each time step, and Wx is a matrix that has different theyights for each input. For each matrix theyight, the magnitude affects the number of hidden state vectors and the current vector or previous state vector affects the hidden state vector. Then, the next process is to enter the last hidden state vector into the binary softmax classifier to produce negative or positive possibility of polarity. The above polarity is generated from the multiplication bettheyen hidden state vectors and the theyight matrix, and then the results are entered into the softmax function.

Results and Analysis

Experimental Word vector (Embedding matrix)

The word2vec model, has a matrix containing 136,281 vector words with dimensions of 300 for each vector. The model will be divided into two parts. The first part is a list of words used in python and the second part is an embedding matrix with dimensions of 136,281x300 which is used to hold all the values of the word vectors. This model collects words that have contexts that are almost as close as together in a vector space table 1.

Table 1. Word Vector

Positive	**Word**	*makan*	*makanan*	Sarapan	*breakfast*
	Vector	0.7336804866790771	0.7336804866790771	0.6317768692970276	0.6317768692970276
Negative	**Word**	*Jorok*	*Kusam*	*Kotor*	*dekil*
	Vector	0.5278632044792175	0.5278632044792175	0.38577190041542053	0.3812659978866577

The next step is to make a representation of each word vector. They use a Tensorflow function to take two arguments. The first argument is used to take the matrix from word2vec and the second argument to retrieve the index of each word. Vector index is an index of rows for each word such as in table 2.

Table 2. Index Word

sentence	*Saya suka kamar luas dekat kolam renang bersih dan segar*									
word	*saya*	*suka*	*Kamar*	*Luas*	*Deket*	*Kolam*	*Renang*	*Bersih*	*Dan*	*segar*
index	56	260	72	291	138	48	49	88	3	1253

Experimental Data Set

They use 5000 data sets to train the model. The data sets is divided into two, i.e. positive and negative. Each data sets is stored in a file with csv extension like in table 3.

Table 3. Review of Data set

	Positive review	**Negative review**
Total File	2,500	2,500
Total words	989,896	1,053,227
Average words	28,99	33,53

Figure 4 is a visual form with histogram format from a data set review. From the histogram, the average number of words is below 50. Then the max sequence value for our model is 50. The average number of sentences in the model is used as the max sequence length

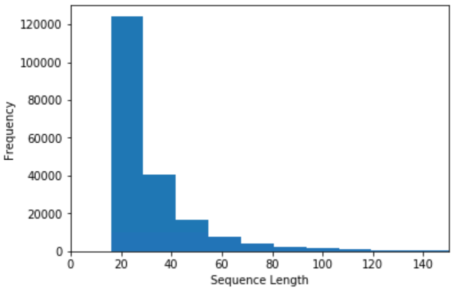

Figure 4. Histogram review data set

They create a set of matrices from the data sets above. RNN requires a collection of these matrices as input. From the training process They get 5000 x 50 matrices. Table 4 shows the example of index vector for a sentence. The data sets sets are then broken down by 75% for the training data sets and 25% for the testing data sets.

Table 4. Data set matrix

Sentence									
Staf sangat bagus, hotel nyaman seperti kita mengambil 3 kamar suite kerajaan dan rungan besar dan memiliki ruang terpisah. Hotel ini bagus, sarapannya enak, Cuma jalanya kecil, cukup untuk satu mobil. Semua baik baik saja									
Index vector									
403	44	185	37	828	146	154	1186	72	395
15900	3	478	329	3	802	491	37	39	185
212	33	45	2477	169	87	85	974	912	32
108	108	482	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

Experiment RNN

They used some hyperparameters for creating RNN model in table 5. Hyperparameter is used to improve the performance of the model.

Table 5. Hyperparameters

No	**Hyperparameter**	**Total**
1	Batch size	24
2	Number of clasess	2
3	Number of training iteration	10000

In this model, They use two placeholders. The first part is used for input into the RNN network while the second part is used for labels. The second part, placeholder labels represent examples of positive and negative training sets.

Result

In the training process, first They accommodate a collection of related reviews and labels. By using the run function, the network can define values where They can optimize components and minimize loss. in addition, the run function is used to feed a collection of reviews and labels. This process will be repeated as many as iterations have been set.

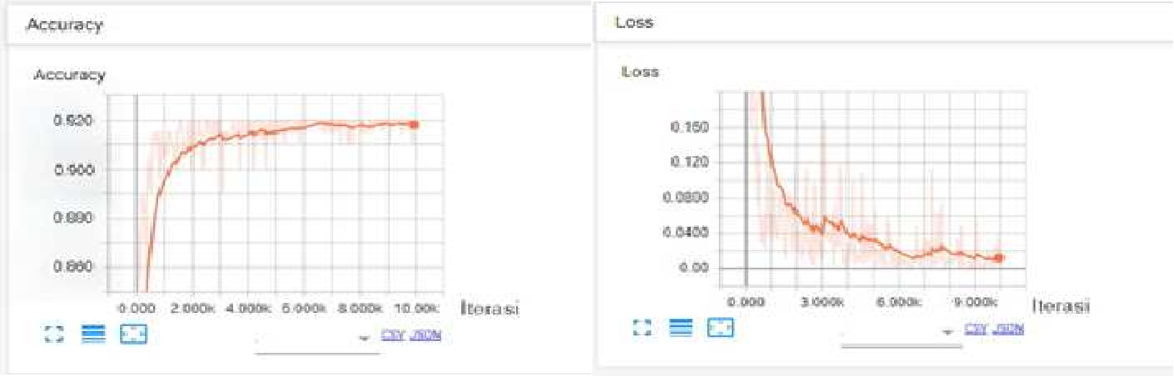

Figure 5. Accuracy and Loss

Figure 5 shows that pretrained model has accuracy approaching 92% percent and loss approaching 0 percent. As a comparison material in this study they also conducted a test using other algorithms such as CNN, Naive Bayes, RNN conv. Accuracy results can be seen in table 6. These results show that RNN with Word2vec can lead to better accuracy rate than the other model.

Table 6. Accuracy

Model	**accuracy**
CNN + Word2vec	0.8923
Naïve Bayes	0.441
RNN Conv	0.8877
RNN + Word2vec	0.9198

Conclusion

In this research they have proposed model for sentiment analysis using RNN and word2vec. RNN in this model is implemented using framework Tensorflow. The research results show that the model approach has better accuracy with other machine learning models with result of accuracy is 91.98%. From this research, They also have to pay attention to the possibility of overfitting the model when carrying out the testing process. In the future They can try to use RNN and LSTM to overcome overfitting problems and to improve the performance of the model.

Reference

[1] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? Sentiment Classification Using Machine Learning Techniques," in *Proceedings of the ACL-02 conference on Empirical methods in natural language processing - EMNLP '02*, 2002, vol. 10, pp. 79–86.

[2] A. Agarwal, B. Xie, I. Vovsha, O. Rambow, and R. J. Passonneau, "Sentiment Analysis of Twitter Data," in *Proceedings of the Workshop on Language in Social Media (LSM 2011)*, 2011, pp. 30–38.

[3] F. Poecze, C. Ebster, and C. Strauss, "Social media metrics and sentiment analysis to evaluate the effectiveness of social media posts," *Procedia Comput. Sci.*, vol. 130, pp. 660–666, 2018.

[4] C. Day, "The Importance of Sentiment Analysis in Social Media Analysis," *LinkedIn*, 2015.

[Online]. Available: https://www.linkedin.com/pulse/importance-sentiment-analysis-social-media-christine-day/. [Accessed: 01-May-2019].

[5] J. Bhaskar, K. Sruthi, and P. Nedungadi, "Hybrid Approach for Emotion Classification of Audio Conversation Based on Text and Speech Mining," *Procedia Comput. Sci.*, vol. 46, pp. 635–643, Jan. 2015.

[6] P. Singhal and P. Bhattacharyya, "Sentiment Analysis and Deep Learning : A Survey," *CoRR*, 2016.

[7] K. Cho, D. Bahdanau, F. Bougares, H. Schtheynk, and Y. Bengio, "Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1724–1734.

[8] B. Liu, Y. Dai, X. Li, W. S. Lee, and P. S. Yu, "Building text classifiers using positive and unlabeled examples," in *Third IEEE International Conference on Data Mining*, pp. 179–186.

[9] G. Li, S. C. H. Hoi, K. Chang, and R. Jain, "Micro-blogging Sentiment Detection by Collaborative Online Learning," in *2010 IEEE International Conference on Data Mining*, 2010, pp. 893–898.

[10] B. Pang and L. Lee, "Opinion Mining and Sentiment Analysis," *Found. Trends® Inf. Retr.*, vol. 2, no. 1–2, pp. 1–135, 2008.

[11] W. Feng, J. Sun, L. Zhang, C. Cao, and Q. Yang, "A support vector machine based naive Bayes algorithm for spam filtering," *2016 IEEE 35th Int. Perform. Comput. Commun. Conf. IPCCC 2016*, pp. 1–8, 2017.

[12] H. Shirani-Mehr, "Applications of Deep Learning to Sentiment Analysis of Movie Reviews," 2015.

[13] A. Timmaraju and V. Khanna, "Sentiment Analysis on Movie Reviews using Recursive and Recurrent Neural Network Architectures," *Semant. Sch.*, pp. 2–7, 2015.

[14] L. Zhang and C. Chen, "Sentiment Classification with Convolutional Neural Networks: An Experimental Study on a Large-Scale Chinese Conversation Corpus," in *2016 12th International Conference on Computational Intelligence and Security (CIS)*, 2016, pp. 165–169.

[15] P. Mishra, R. Rajnish, and P. Kumar, "Sentiment analysis of Twitter data: Case study on digital India," in *2016 International Conference on Information Technology (InCITe) - The Next Generation IT Summit on the Theme - Internet of Things: Connect your Worlds*, 2016, pp. 148–153.

[16] M. Thomas and L. C.A, "Sentimental analysis using recurrent neural network," *Int. J. Eng. Technol.*, vol. 7, no. 2.27, p. 88, 2018.

[17] F. Miedema, "Sentiment Analysis with Long Short-Term Memory networks," 2018.

[18] X. Ouyang, P. Zhou, C. H. Li, and L. Liu, "Sentiment Analysis Using Convolutional Neural Network," in *2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing*, 2015, pp. 2359–2364.

[19] J. Panthati, J. Bhaskar, and T. K. Ranga, "Sentiment Analysis on Customer Reviews using Deep Learning," *Int. J. Comput. Sci. Eng.*, vol. 6, no. 7, pp. 1023–1024, 2018.

[20] P. Milev, "Conceptual Approach for Development of Theyb Scraping Application for Tracking Information," *Econ. Altern.*, no. 3, pp. 475–485, 2017.

[21] X. Rong, "word2vec Parameter Learning Explained," Nov. 2014.

[22] A. Deshpande, "Perform sentiment analysis with LSTMs, using TensorFlow - O'Reilly Media." [Online]. Available: https://www.oreilly.com/learning/perform-sentiment-analysis-with-lstms-using-tensorflow. [Accessed: 20-Apr-2019].