Jurnal Informatika Universitas Pamulang **ISSN: 2541-1004**
Penerbit: Program Studi Teknik Informatika Universitas Pamulang **e-ISSN: 2622-4615**
Vol. 8, No. 2, Juni 2023 (279-286) **10.32493/informatika.v8i2.32210**

Sentiment Analysis of the 2024 Presidential Candidates Using SMOTE and Long Short Term Memory

Christian Sri Kusuma Aditya[1], Galih Wasis Wicaksono[2], and Hilman Abi Sarwan Heryawan[3]

[1,2,3]Informatics Department, Universitas Muhammadiyah Malang, Jl. Raya Tlogomas No.246 Malang, Jawa Timur, Indonesia, 65144
e-mail: [1]christianskaditya@umm.ac.id, [2]galih.w.w@umm.ac.id, [3]hilmanabish@gmail.com

Submitted Date: June 09th, 2023 Reviewed Date: June 20th, 2023
Revised Date: June 26th, 2023 Accepted Date: June 30th, 2023

Abstract

Numerous political leaders participate in elections since they are a crucial component of the political process. Since electability is an issue, steps are taken to make political candidates running in general elections more electable. The media, including internet news media, has emerged as one of the key strategies for raising electability. Reader comments can be analyzed for sentiment to provide an evaluation of political figures. However, because the comments contain unstructured content, particularly in Indonesian text, it is difficult to interpret the sentiments of different comments in online news media. In this research, an analysis of public sentiment towards the 2024 presidential candidates will be carried out which is expressed through the Twitter social network. There are several stages to carry out sentiment analysis, including the stages of data collection, data preprocessing, balancing the distribution of the number of datasets, and sentiment classification using the LSTM method with word2vec feature representation. The results of this study show that the LSTM method combined with SMOTE due to the limited amount of data is able to produce a fairly good LSTM model with an average accuracy of 89.42% and a loss value of 0.24, the ideal scenario is when the accuracy is high and the loss is minimal, in which case the LSTM model only exhibits minor errors on a subset of the data.

Keywords: Sentiment; Twitter; SMOTE; LSTM; Word2vec, Presidential, 2024

1 Introduction

Presidential elections in Indonesia's national history have been held for several times, but general elections carried out directly by the people of Indonesia have only started for the first time in the reform era after the New Order era collapsed, namely in 2004. The presidential election which will be held in 2019 is an exciting moment. important for realizing democracy in the Unitary State of the Republic of Indonesia. Candidates and successful teams in the 2019 presidential election can utilize social media to convey campaign messages (Badrika, 2018), one of the media that is actively used for campaigns is Twitter. In the upcoming 2019 presidential election, as many as 40% or around 90 million people are first-time voters or the millennial generation, so the power of social media cannot be underestimated for the electability of the two pairs of candidates.

Social networks such as Twitter are now a very popular communication tool among cyberspace users and can be used as a campaign media for presidential election participants in conveying a positive image for their respective spouses to prospective voters and their supporters. 77% of Indonesia's Twitter users are reportedly active users, according to data given by Twitter Indonesia at the end of 2016. This is evident from the massive volume of tweets sent and received in 2016—4.1 billion tweets, to be exact (Alsaeedi, 2019). According to a Twitter sentiment analysis, a tweet's text either conveys negative, positive, or neutral sentiments. It is a text analysis employing machine learning and natural language processing (NLP). Twitter is often used as a means to express opinions, so the data generated by Twitter is very useful for processing because it contains crucial information when analyzed. There were many positive and negative comments from the public

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License
Copyright © 2023 Christian Sri Kusuma Aditya, Galih Wasis Wicaksono, Hilman Abi Sarwan Heryawan

before the election was held or during the election regarding the election being held.

In the process of sentiment analysis there are several Naïve Bayes methods, Maximum Entropy, SVM (Fachrurrozi & Yusliani, 2015), Neural Network (Ito, 2018), previous research Twitter data was used for sentiment analysis of the 2017 DKI Pilkada opinion using the Naïve Bayes method resulting in the highest accuracy value of 74.81% (Lestari, 2017).

An algorithm that is frequently used in classification methods is the Naive Bayes classifier. The most suitable alternative is chosen using this procedure, which applies probability theory (Bramer, 2013). Another research (Kurniawan & Susanto, 2019) regarding sentiment analysis for the 2019 Presidential Election (Pilpres), applied the k-means method to divide a set of tweet data into a number of specific groups which were then followed by Naïve Bayes modeling. From the results of testing 100 and 150 test data in this study, it was obtained an average accuracy of 93.35% and an average error rate of 6.66%.

Deep learning techniques have started to be used in sentiment analysis research in addition to the machine learning approach outlined above. The machine learning technique previously employed has various drawbacks, including training processing times that tend to be long and poor accuracy, which is one reason why deep learning is becoming more and more popular (Adam & Josh, 2017).

Other studies have also conducted sentiment analysis using deep learning methods (Kim, 2014). In addition, this study also used pre-trained word2vec to create word-level embedding (word vectors) which were used as input for the CNN model that was built. The best accuracy results obtained were 88.1% using the CNN-Multichannel model on the Stanford sentiment 2 class dataset.

Previously there were many studies on sentiment analysis which had also been carried out, namely to review novels using the Long Short-Term Memory (LSTM) method. Based on the results of the tests conducted, it shows that the LSTM method has an accuracy of 72.85%, precision of 73%, recall of 72%, and f-measure of 72%. This study compared the LSTM with the results of the accuracy of the Naïve Bayes method with an accuracy value of 67.88%, 69% precision, 68% recall, and 68% f-measure. This study showed that the performance of the LSTM method was better than the Naïve Bayes method (Nurrohmat, 2019).

LSTM or Long Short-Term Memory is an updated model of the RNN model which is used to manage sequential data by storing the results of previous information (Firmansyah, 2020). LSTMs can handle variable length sequences and bidirectional input, which is useful for NLP tasks such as machine translation, text generation, and sentiment analysis. RNNs are simpler and faster to train than LSTMs, because they have fewer parameters and computations (Ivanedra & Mustikasari, 2019).

Based on the description of the literature above, this study will develop a modeling with a deep learning approach in conducting sentiment analysis of the 2024 presidential candidates, and also uses word2vec as word vector forming (Putri, 2022). Due to the unbalanced number of class compositions or news categories, integration is also carried out with the Synthetic Minority Over-sampling Technique (SMOTE) (Tannady, 2022), where SMOTE works by generating synthetic samples from the minority class by connecting close neighbors in the feature space (Widhiyasana, 2021). Thus, this technique is able to create synthetic data that describes the variation and complexity of minority classes, thereby reducing bias and increasing overall classification accuracy.

2 Research Methods

The research method that will be carried out in conducting sentiment analysis regarding the 2024 presidential candidates using LSTM is shown in Figure 1.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License
Copyright © 2023 Christian Sri Kusuma Aditya, Galih Wasis Wicaksono, Hilman Abi Sarwan Heryawan

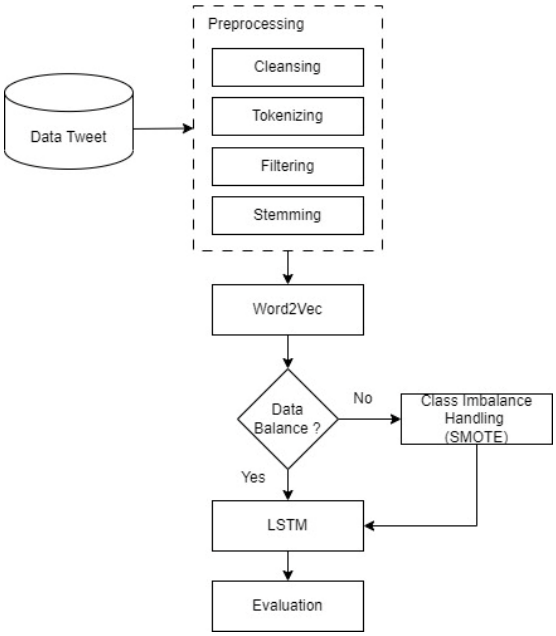

Figure 1. Research Design Diagram

2.1 Dataset

Data collection is a process to obtain documents that will be used as input to the system. Data obtained from Twitter utilizes the Twitter API using Python programming with the Tweepy library to retrieve tweet data. This data collection will be based on predetermined search (keywords), namely the two strongest candidates for the presidential candidate for the 2024 - 2029 period. The three presidential candidates to be analyzed for sentiment are Ganjar Pranowo, Prabowo Subianto dan Anies Baswedan. The number of tweets that were collected for the three candidates was 1903.

2.2 Preprocessing

Preprocessing is the initial stage in processing text data so that sentiment analysis can be carried out. The first stage of preprocessing used is cleansing, whereby cleaning the document from punctuation marks, URL (Uniform Resource Locator) addresses, and making all letters uniform using lowercase letters. The second stage is Tokenizing, which is a process carried out to break a sentence into several parts or words. The third stage is Filtering, which is the process of removing words that often appear because they are considered meaningless by iterating over each list of tokenizing results. The fourth stage is Stemming, which is the process of decomposing or mapping a word into its basic form. The stemming algorithm used in this study uses Enhanced Confix Stripping (Denny & Spirling, 2018)(Hickman, 2022). The results of the dataset that has gone through the preprocessing stage can be seen in Table 1.

Table 1. Result of Preprocessing

Preprocessing Results	Category of Presidential Candidates
[1] jika; pilpres; 2024; laksana; jujur; adil; cawe-cawe; siapa; menang; [2] prabowo; ubah; jokowi; msih; tanggap; santai; [3] hati; politik; jokowi; halus; tdk; bs; tebak; tpi; telak; mengena; surya; partai; [4] masyarakat; lihat; sebab; gak; sekedar; tegas; keren; bareng; pak; prabowo; fokus; juang; rakyat;	Prabowo Subianto
cocok; lanjut; kinerja; pak; jokowi; terus; maju; bersama; ganjar; pranowo; pak; ganjarpranowo; kan; suka; jogging; pasti; siap; nanti; jalan; sibuk; sebagai; orang; presiden; video; mimin; jdi; semakin; yakin; kalau; rambut; putih; ganjarpranowo; memang; pemimpin; yg; dengar; aspirasi; rakyat;	Ganjar Pranowo
mengapa; jokowi; kalau; anies; baswedan; menang; presiden; simak; buruh; transportasi; dukung; anies; pilpres; 2024; saat; dukung; presiden; yg; milik; rekam; jejak; prestasi; memukau; akun; resmi; anies; baswedan; warga; DKIJakarta; kelola; tim; kicau; pribadi; tanda; abw; sudah; mantan; cinta; pemimpin; amanah; begitu	Anies Baswedan

2.3 Synthetic Minority Oversampling Technique (SMOTE)

Working with imbalanced datasets is challenging since most machine learning algorithms underperform on the minority class, despite the fact that this class' performance is frequently crucial (Fernández, 2018). In SMOTE, examples that are close to one another in the feature space are chosen, a line is drawn between them, and a new sample is then drawn at a location along the line (Pan, 2020).

More specifically, a representative from the minority class is originally chosen at random. The next step is to find the example's k nearest neighbors (k is typically equal to 5). Between two

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License
Copyright © 2023 Christian Sri Kusuma Aditya, Galih Wasis Wicaksono, Hilman Abi Sarwan Heryawan

instances and each instance's randomly chosen neighbor, a synthetic example is built in feature space (Camacho, 2020). The results of changing the amount of data distribution can be seen in Figure 2.

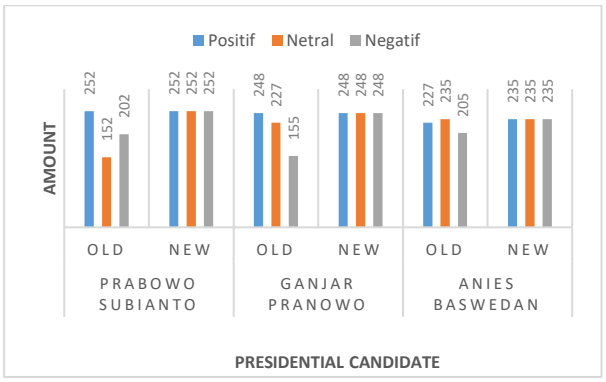

Figure 2. SMOTE Result

2.4 Word2Vec

Word2Vec is a word embedding algorithm that maps every word in text into vectors and is widely used in NLP (Natural Language Processing) research. Word2Vec converts words into vectors that represent their semantic meaning. (Jatnika, 2019).

Word2vec uses vectors to disseminate the numerical representation of word properties, such as word context. These vectors are obtained by Word2Vec using a neural network. Only 3 layers make up the Word2vec architecture: input, projection (hidden layer), and output. A one-hot encoded vector with a length equal to the number of distinct words in the training data serves as the input to Word2vec. The "Skip-gram" and "Continous Bag of Word" (CBOW)" types of neural network designs from Word2Vec are available. In contrast to the skip-gram variety, which uses a target word and tries to forecast the context words around it, the CBOW (continuous bag of words) form uses a set of context words and tries to predict a target word (Grohe, 2020). In this research will use the skip-gram variant, the illustration shown in Figure 3.

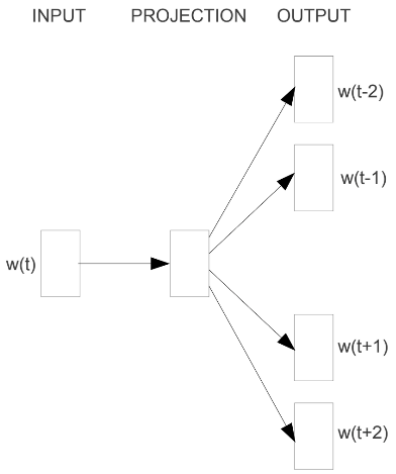

Figure 3. Illustration of the Skip-gram

2.5 Long Short Term Memory (LSTM)

In neural networks, the Long Short-Term Memory (LSTM) structure can be applied. Recurrent neural networks (RNNs) of this kind anticipate input in the form of a series of features. For data like time series or long strings of text, it is helpful (Yu, 2019).

Gates refer to these three LSTM unit components. They control the flow of information into and out of the memory cell or lstm cell. The Forget gate, shown in equation 1, the Input gate, shown in equation 2, and the Output gate, shown in equation 3, are the three gates. With each neuron having a hidden layer and a present state, a layer of neurons in a standard feedforward neural network can be thought of as an LSTM unit composed of these three gates plus a memory cell (Sherstinsky, 2020). An illustration showing the LSTM architecture can be seen in Figure 4.

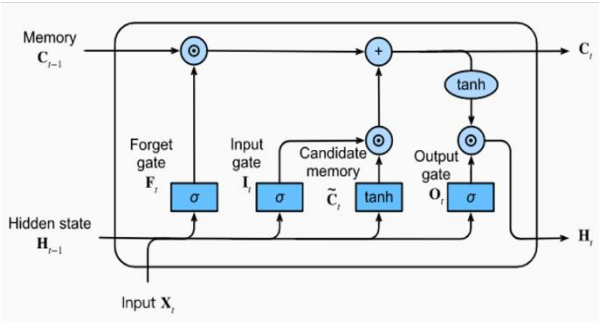

Figure 4. LSTM Architecture

To maintain or reject the data from the previous time step is the first choice made in a cell of the LSTM neural network. Below is the forget gate equation.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License
Copyright © 2023 Christian Sri Kusuma Aditya, Galih Wasis Wicaksono, Hilman Abi Sarwan Heryawan

Forget Gate

$$f_t = \sigma\left(x_t * U_f + H_{t-1} * W_f\right) \quad (1)$$

Input Gate

$$i_t = \sigma(x_t * U_i + H_{t-1} * W_i) \quad (2)$$

Output Gate

$$o_t = \sigma(x_t * U_o + H_{t-1} * W_o) \quad (3)$$

x_t: input to the current timestamp.

U_f: weight associated with the input

U_i: weight matrix of input

U_o: weight matrix of output

H_{t-1}: hidden state of the previous timestamp

W_f: weight matrix associated with the hidden state

σ: sigmoid

Later, a sigmoid function is applied to it. That will make f_t a number between 0 and 1. The cell state of the preceding timestamp is later multiplied by this f_t, as seen below. The equation from sigmoid function can be seen at equation 4 and 5.

$$C_{t-1} * f_t = 0 \ldots if\ f_t = 0\ (forget\ everything) \quad (4)$$

$$C_{t-1} * f_t = C_{t-1} \ldots if\ f_t = 1\ (forget\ nothing) \quad (5)$$

Equation 6 represents the new data that has to be sent to the cell state and is a function of a concealed state at timestamp $t-1$ in the past and input x at timestamp t.

The activation function here is $tanh$. The $tanh$ function causes the value of fresh information to range from -1 to 1. If the value of N_t is negative, the information is subtracted from the cell state, and if the value is positive, The data is inserted at the present timestamp to the cell state.

$$N_t = tanh(x_t * U_c + H_{t-1} * W_c)\ (new\ information) \quad (6)$$

However, the N_t won't be added directly to the cell state. Here comes the updated equation 7.

$$C_t = f_t * C_{t-1} + i_t * N_t\ (updating\ cell\ state) \quad (7)$$

Here, C_{t-1} is the cell state at the current timestamp, and the others are the values we have calculated previously.

2.6 Evaluation

A performance indicator for machine learning categorization is the confusion matrix. The rows of the matrix correspond to the instances in each actual class, whereas the columns correspond to the examples in each anticipated class, or vice versa.

In the confusion matrix table, there are four values that are produced: True Positive (TP), False Positive (FP), False Negative (FN), and True Negative (TN). Equation 8 illustrates the accuracy calculation process.

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+FP+FN+TN}} \quad (8)$$

In a machine learning model, equation 9 will assess the error between the anticipated and actual values in addition to calculating accuracy. How well your algorithm models your dataset can be assessed using the Loss function (Shutaywi, 2021). A good model is one where the loss function value is lower; otherwise, we must adjust the model's parameters to reduce loss.

$$\text{Loss} = -\sum_{j=1}^{k} y_j \log\left(\hat{y}_j\right) \quad (9)$$

k : number of class in the data
j : training example in a data set
y_j : ground truth label for ith training example
$\hat{y}_j$: prediction for ith training example

3 Results and Discussions

3.1 Classification Results with LSTM

In the first trial scenario, after preprocessing the data, to get a model with good performance, regularization will be carried out with adjusted hyperparameter values, as shown in Table 2. The following hyperparameter values quote from previous studies that also carry out sentiment analysis using LSTM.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License
Copyright © 2023 Christian Sri Kusuma Aditya, Galih Wasis Wicaksono, Hilman Abi Sarwan Heryawan

ISSN: 2541-1004
e-ISSN: 2622-4615
10.32493/informatika.v8i2.32210

Table 2. Modeler Hyperparameters

Jenis	Nilai
Units	150
Dropout	0.221
Activation	Relu
Optimizer	Adam
Epoch	150
Batch	50

The process of modeling the dataset is done by splitting the dataset into two parts, namely 80% for training data and 20% for testing data. Testing is done on the value of Learning rate (Lr). The training data is trained with the LSTM modeling process, then the model performance results are evaluated using accuracy measurements and the value of the categorical cross entropy loss function.

Table 3. Result of Preprocessing

	Lr	Accuracy (%)	Loss
Prabowo	0.01	87.23	0.33
Subianto	0.001	89.31	0.21
Ganjar	0.01	88.22	0.27
Pranowo	0.001	89.18	0.22
Anies	0.01	91.67	0.25
Baswedan	0.001	90.91	0.18

It can be seen in the Table 3, the test produced a fairly good average accuracy, with the highest score in the Anies Baswedan candidate category with an accuracy value of 91.67% and a Loss value of 0.25. For the presidential candidate category Prabowo Subianto has a slightly lower difference accuracy value than the average, this is possible because the amount of initial data before the SMOTE process also has a smaller amount when compared to the amount of data in the categories of presidential candidates Ganjar Pranowo and Anies Baswedan. In the ideal scenario, the LSTM model would only produce modest errors on a small portion of the data because to high accuracy and minimal loss.

3.2 Effects of Applying SMOTE

In the second scenario, due to the imbalance in the distribution of the number of datasets possessed, testing is done to determine the impact of employing SMOTE. The issue with imbalanced categorization is that there aren't enough examples of the minority class for a model to learn the decision boundary effectively. Instances from the minority class can be oversampled as one approach to resolving this issue. The average of accuracy in LSTM modeling has grown by 9.69%, as shown in Figure 5. This proves that the more data train used, the better the results obtained.

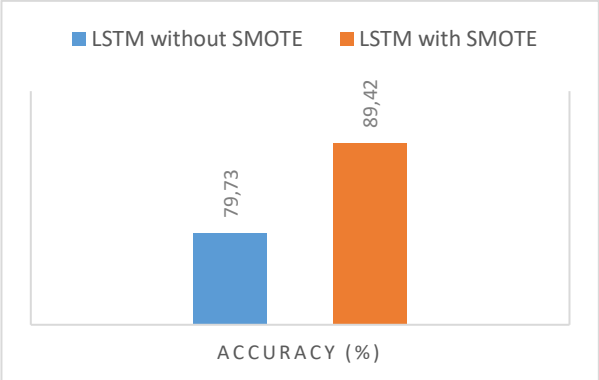

Figure 5. Comparison of the Effects of Using SMOTE

if the amount of training data is not large, underfitting will occur so that the performance of a model is not optimal. because it is exceedingly expensive to obtain data or because there aren't many samples taken routinely. In that instance, using data augmentation may be a good option.

4 Conclusion

Based on the results of applying the LSTM with word2vec and SMOTE using a dataset taken from Twitter about 2024 Presidential Candidates for sentiment analysis tasks provides an average accuracy value of 89.42%. In this instance, it also demonstrates how the SMOTE approach can improve the correctness of modeling designs created from unbalanced datasets. Due to the learning algorithm's lack of sufficient training data, accuracy on test and training data may be subpar. The average accuracy in LSTM modeling with SMOTE grows by 9.69% when compared to LSTM without SMOTE.

5 Future Work

Because the use of a model with a deep learning approach has a fairly high average accuracy value, future research will compare other deep learning models besides LSTM, for example RNN and CNN, to compare the accuracy values of each model.

References

A. J. Putri, A. S. Syafira, M. E. Purbaya, and D. Purnomo, “Analisis Sentimen E-Commerce Lazada pada Jejaring Sosial Twitter Menggunakan Algoritma Support Vector

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License
Copyright © 2023 Christian Sri Kusuma Aditya, Galih Wasis Wicaksono, Hilman Abi Sarwan Heryawan

Machine," Jurnal TRINISTIK: Jurnal Teknik Industri, Bisnis Digital, dan Teknik Logistik, vol. 1, no. 3, pp. 16–21, Mar. 2022, doi: 10.20895/trinistik.v1i1.447.

Alsaeedi, A., & Khan, M. Z. (2019). A study on sentiment analysis techniques of Twitter data. International Journal of Advanced Computer Science and Applications, 10(2), 361-374.

A. R. T. Lestari, R. S. Perdana dan M. A. Fauzi, "Analisis Sentimen Tentang Opini Pilkada Dki 2017 Pada Dokumen Twitter Berbahasa Indonesia Menggunakkan Naive Bayes dan Pembobotan Emoji," Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, vol. 1, pp. 1718-1724, 2017

Badrika, A., Sulandari, S., & Astawa, I. W. (2022). IMPLEMENTASI PERATURAN KOMISI PEMILIHAN UMUM NOMOR 23 TAHUN 2018 TENTANG KAMPANYE PEMILIHAN UMUM TAHUN 2019 DI KABUPATEN GIANYAR. Jurnal Ilmiah Cakrawarti, 5(2), 80-89.

Camacho, L., Douzas, G., & Bacao, F. (2022). Geometric SMOTE for regression. Expert Systems with Applications, 193, 116387.

Denny, M. J., & Spirling, A. (2018). Text preprocessing for unsupervised learning: Why it matters, when it misleads, and what to do about it. Political Analysis, 26(2), 168-189.

Fernández, A., Garcia, S., Herrera, F., & Chawla, N. V. (2018). SMOTE for learning from imbalanced data: progress and challenges, marking the 15-year anniversary. Journal of artificial intelligence research, 61, 863-905.

Firmansyah, M. R., Ilyas, R., & Kasyidi, F. (2020, September). Klasifikasi Kalimat Ilmiah Menggunakan Recurrent Neural Network. In Prosiding Industrial Research Workshop and National Seminar (Vol. 11, No. 1, pp. 488-495).

G. Adam and P. Josh, Deep Learning: A Practitioner's Approach. 2017.

Grohe, M. (2020, June). word2vec, node2vec, graph2vec, x2vec: Towards a theory of vector embeddings of structured data. In Proceedings of the 39th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems (pp. 1-16).

Herman, "Indonesia Masuk Lima Besar Pengguna Twitter," 03 05 2017. [Online]. Available: http://www.beritasatu.com/iptek/428591-indonesia-masuk-lima-besar-pengguna-twitter.html. [Diakses 2018 04 15]

Herremans, D., & Chuan, C. H. (2017). Modeling musical context with word2vec. arXiv preprint arXiv:1706.09088.

Hickman, L., Thapa, S., Tay, L., Cao, M., & Srinivasan, P. (2022). Text preprocessing for text mining in organizational research: Review and recommendations. Organizational Research Methods, 25(1), 114-146.

Ivanedra, K., & Mustikasari, M. (2019). Implementasi Metode Recurrent Neural Network Pada Text Summarization Dengan Teknik Abstraktif. J. Teknol. Inf. dan Ilmu Komput, 6(4), 377.

Ito, T., Tsubouchi, K., Sakaji, H., Yamashita, T., & Izumi, K. (2020). Contextual sentiment neural network for document sentiment analysis. Data Science and Engineering, 5, 180-192.

Jatnika, D., Bijaksana, M. A., & Suryani, A. A. (2019). Word2vec model analysis for semantic similarities in english words. Procedia Computer Science, 157, 160-167.

Kurniawan, I., & Susanto, A. (2019). Implementasi Metode K-Means dan Naive Bayes Classifier untuk Analisis Sentimen Pemilihan Presiden (Pilpres) 2019. Jurnal Eksplora Informatika, 9(1), 1-10.

M. A. Nurrohmat and A. SN, "Sentiment Analysis of Novel Review Using Long Short-Term Memory Method," IJCCS (Indonesian J. Comput. Cybern. Syst., vol. 13, no. 3, p. 209, 2019, doi: 10.22146/ijccs.41236

M. Bramer, "Principles of Data Mining. Undergraduate Topics in Computer Science," Ch. 12: Estimating the Predictive Accuracy of a Classifier, Nov. 2013

M. Fachrurrozi dan N. Yusliani, "Analisis Sentimen Pengguna Jejaring Sosial Menggunakan Metode Support Vector Machine," Konferensi Nasional Sistem Informasi, vol. 1, no. Konferensi Nasional Sistem Informasi, 2015.

Pan, T., Zhao, J., Wu, W., & Yang, J. (2020). Learning imbalanced datasets based on SMOTE and Gaussian distribution. Information Sciences, 512, 1214-1233.

Sherstinsky, A. (2020). Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. Physica D: Nonlinear Phenomena, 404, 132306.

Shutaywi, M., & Kachouie, N. N. (2021). Silhouette analysis for performance evaluation in machine learning with applications to clustering. Entropy, 23(6), 759.

Tannady, S. M. N., Setiabudi, D. H., & Tjondrowiguno, A. N. (2022). Penerapan Long-Short Term Memory dengan Word2Vec Model untuk Mendeteksi Hoax dan Clickbait News pada Berita Online di Indonesia. Jurnal Infra, 10(2), 28-34.

Widhiyasana, Y., Semiawan, T., Mudzakir, I. G. A., & Noor, M. R. (2021). Penerapan Convolutional Long Short-Term Memory untuk Klasifikasi Teks Berita Bahasa Indonesia. Jurnal Nasional

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License
Copyright © 2023 Christian Sri Kusuma Aditya, Galih Wasis Wicaksono, Hilman Abi Sarwan Heryawan

Teknik Elektro dan Teknologi Informasi| Vol, 10(4).

Y. Kim, "Convolutional neural networks for sentence classification," EMNLP 2014 - 2014 Conf. Empir. Methods Nat. Lang. Process. Proc. Conf., pp. 1746–1751, 2014, doi: 10.3115/v1/d14-1181.

Yu, Y., Si, X., Hu, C., & Zhang, J. (2019). A review of recurrent neural networks: LSTM cells and network architectures. Neural computation, 31(7), 1235-1270.

Zhang, Y., Tiwari, P., Song, D., Mao, X., Wang, P., Li, X., & Pandey, H. M. (2021). Learning interaction dynamics with an interactive LSTM for conversational sentiment analysis. Neural Networks, 133, 40-56.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) License

Copyright © 2023 Christian Sri Kusuma Aditya, Galih Wasis Wicaksono, Hilman Abi Sarwan Heryawan