Comparison of Word2vec and Doc2vec Methods for Text Classification of Product Reviews

2022 6th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE) | 979-8-3503-9961-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICITISEE57756.2022.10057702

Ivan Rifky Hendrawan
Magister of Informatic Engineering
Universitas Amikom Yogyakarta
Yogyakarta,Indonesia
ivanrifky@students.amikom.ac.id

Ema Utami
Magister of Informatic Engineering
Universitas Amikom Yogyakarta
Yogyakarta, Indonesia
ema.u@amikom.ac.id

Anggit Dwi Hartanto
Magister of Informatic Engineering
Universitas Amikom Yogyakarta
Yogyakarta, Indonesia
anggit@amikom.ac.id

***Abstract*—Word Embedding is the optimal tool for many Natural Language Processing (NLP) tasks, especially those that require native input as a text feature. In this study, we will try to compare the performance of Word2vec and Doc2vec on unbalanced review text data using XGBoost, which in the end will look for which combination is suitable for processing unbalanced data Word2vecProduces high-performance values. Doc2vec has a better advantage over Word2vec with the condition that we will know the closeness between the review sentences, not the closeness of words anymore. Based on experiments, the performance of Word2vec and Doc2vec paired with the XGBoost Algorithm was able to classify unbalanced datasets with an average F1 Score value of 0.9342 for Word2vec and 0.9344 for Doc2vec. The results of testing the vector dimensions for both Word2vec and Doc2vec show that the larger the size of the word vector dimensions used, the longer the training time required.**

Keywords—Word2vec,Doc2vec,XGBoost,Natural Language Processing

I. Introduction

Word Embedding is the optimal tool for many NLP tasks, especially those that require native input as a text feature[1]. There are different types of models for constructing Word Embedding, and each has its advantages and disadvantages. According to Suleiman(2019), Word Embedding is a form of text representation that uses vectors in such a way that words with the same meaning and syntax will be given the same vector meaning[2]. Word embedding can be considered a textual feature, so it can be counted as a pre-processing step in more advanced NLP tasks. This representation is so important because it will have a significant impact on the accuracy or performance of the learning model built or trained [3]. One of the functions of word embedding is being able to represent words better than other traditional Word Embeddings such as Bag-of-word or TF-IDF. Similarities between words can be measured in various ways, one of which is the Euclidean distance. Recently, a two-word insertion model was proposed which plays an important role in various natural language processing applications called the Word2vec model [4] and the Doc2vec model [5].

Word2vec is a shallow neural network model that converts word representation which is a combination of alphanumeric characters into vectors[6]. The vector representation has a relationship property to related words through the training process while for the extraction feature of Doc2vec, unlike Word2vec which can create vector representations of words while taking into account context, Doc2vec can create vector representations of documents [7]. Word2vec is very suitable for use in conditions of large amounts of data. Word2vec has many advantages compared to other word representation methods, such as a faster training process, and more efficiency can handle large-scale datasets [8]. Based on experiments, crucial decisions that can affect the performance of Word2vec are the choice of the model used, the size of the vector, the sub-sampling rate, and the training window. As for Doc2vec, this method can perform feature extraction by using all the information or words in the document[9], because every word in the document is used for the learning process. Doc2vec generates document vectors and word vectors in the training data. Each document in the training data will be represented in word sets and tags [10].

In this research, the word embedding model will be tested on unbalanced data. Classification of data with unbalanced classes is a major problem in the field of machine learning and data mining, for example in e-commerce review problems [11], and text classification problems [12]. When working with unbalanced data, most classifications will result in higher accuracy for large classes than for small classes [13]. This difference is an indicator of poor classification performance. Research has also experienced this imbalance data problem[14] so it affects the recall, precision, and f1-score values. In optimizing classification on unbalanced datasets, Feature extraction is one of the keys to text sentiment analysis, and the appropriate algorithm has an important effect on the results.

Feature extraction is a measurement reduction process that converts the original data into a data set. Functional extraction is effective in reducing the amount of data to be processed while retaining the relevant data from the original data set. Feature extraction can reduce unnecessary information in the data set and also speed up the machine-learning process. To optimize imbalanced data at the classification stage, the XGBoost algorithm can be used, where this algorithm is a popular algorithm for classifying unbalanced data [13]. It is said to be popular in classifying imbalanced data because the boosting method is used to build a new model that predicts errors from the previous model. New models are added until the error correction is no longer possible. By using gradient descent to minimize errors when building a new model, this algorithm is called gradient enhancement. This is supported by research that has been done previously XGBoost produces 96.24% accuracy with datasets collected from the Google Play Store using the Google Play scraper library in

Python totaling 12,969 review data from January 1 – September 30, 2021.

From the problems that have been described previously, In this study, we will try to compare the performance of Word2vec and Doc2vec on unbalanced review text data using XGBoost, which in the end will look for which combination is suitable for processing unbalanced data.

II. STUDY OF LITERATURE

A. Natural Language Processing

Natural Language Processing (NLP) is an application of machine learning and computational techniques that can understand and represent spoken and written texts [15]. According to [16] the NLP model is trained using formal language and according to rules. Two approaches can be taken to process text from social media, the first is to train it repeatedly or the second is to normalize the text.

B. Sentiment Analysis

Sentiment analysis is one of the fields of science that is used to determine public perceptions, whether positive, negative or neutral towards figures, organizations, and issues that are currently happening. Furthermore [17]explains that the basic work in sentiment analysis is to classify the polarity of a document, sentence, or text, whether the opinion expressed in the text is positive, negative, or neutral.

C. Text Preprocessing

Preprocessing is one of the stages in preparing data before it is used in a model. Pre-processing of sentiment analysis is one of the important phases of data in the mining process because the data used in the mining process is not always in an ideal state for processing[18]. Pre-processing techniques that can be used on text sourced from social media include removing usernames, hashtags, URLs, punctuation, repeated letters in a word, excessive spaces, stopwords, and duplicate tweets.

D. Doc2vec

Word embedding is an NLP technique that converts a basic word into a real-valued vector [4]. Doc2vec is a model for representing the numeric value of a document, no matter how old the document is. Research [9]reveals the reason for using this model is that documents are not like words with logical structures. The Doc2vec vector can be used for several purposes, such as finding similarities between sentences/paragraphs [19].

E. Word2vec

Word2vec represents words into vectors that can carry the semantic meaning of the word. This word insertion model is an unsupervised learning application that uses a neural network consisting of a hidden and fully connected layer [3]. The dimension of the weight matrix for each layer is the number of words in the corpus multiplied by the number of hidden neurons in the hidden layer. The Word2vec can be calculated as :

$$H= X * W_1 \tag{1}$$

As H is the hidden layer, X is the previous input neuron and W1 is the weight, the output of the hidden neuron is H which is represented from the second row of the matrix W_1. The weight matrix in the hidden layer of the trained model is used to transform words into vectors.

F. XGboost

Extreme Gradient Boosting (XGBoost) is a machine learning technique for regression analysis and classification based on the Gradient Boosting Decision Tree (GBDT). Study [13]Extreme Gradient Boosting is a decision tree-based ensemble machine learning algorithm that uses a gradient enhancer framework. The ensemble classifiers approach adopts several learning algorithms to get better performance [20]. The XGBoost can be calculated as :

$$\hat{y}_i^{(t)} = \sum_{k=1}^{t} f_k(x_i) \tag{2}$$

For the XGboost algorithm, it is important to determine the number of trees and the depth. The problem in determining the optimal algorithm can be changed by looking for a new classification that can reduce the loss function

G. Confusion Matrix

According to[21]Confusion matrix is a calculation table based on the evaluation of the classification model's performance based on the number of correctly and incorrectly predicted study items.

III. METHOD

Based on the background described, this research will use seven stages of research, which can be seen in Fig. 1.

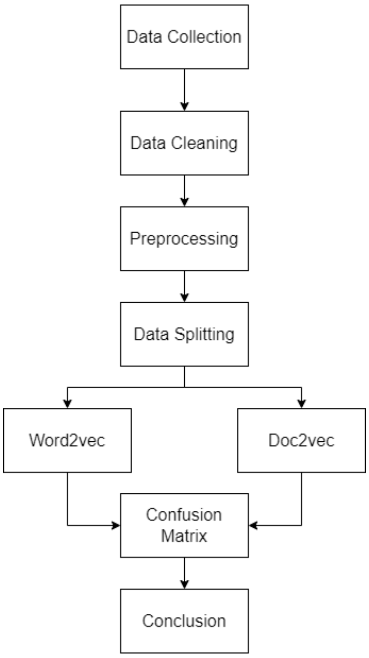

Fig. 1. Research Flow

In Fig. 1 it can be explained that the implementation in this study begins with data collection, where review and rating data are collected on e-commerce applications using Scraping techniques with a platform. Google Collaboratory and Python as the language. Then after the dataset has been collected through Scraping will enter the data cleaning stage, where at this stage two processes are carried out, namely removing irrelevant reviews and removing duplicate reviews. Before being processed and analyzed the dataset will enter the preprocessing process, this stage is useful for ensuring the quality of the data is good before being used during data analysis. Data that has been completed in pre-processing will be divided into two parts, namely testing and training data. In data sharing, distribution scenarios will be used, namely 80:20. The next stage after data sharing is feature extraction where there are two methods used, namely Word2vec and Doc2vec.

The vector dimensions used based on Word2vec and Doc2vec are 50, 100, 200, and 300. The experimental scenario will use these dimensions because this dimension test has been carried out in previous research where the use of dimension variations aims to obtain size dimensions that can produce the best accuracy[22]. The smaller the dimensions the more information is used and discarded. It can help in the process of sentiment recognition. After that is the classification stage, the model that will be used for classification training is XGBoost. In this training phase, we will use two scenarios. The former uses Word2vec + XGBoost and Doc2vec + XGBoost. The next stage is the Confusion matrix, at this stage, an evaluation of the model evaluation is carried out based on the number of study items that are predicted to be true and false. At this stage, the level of accuracy, precision, recall, and f1-score can be known. The final stage of this research is drawing conclusions, at this stage, it will be compared which model has the highest F1 Score value by using a confusion matrix.

IV. ANALYSIS & DISCUSSION

The data used in this study is in the form of Scrapingdata from e-commerce applications. The data scrapingprocess produces 25.581 data. The scrapingprocess is carried out automatically on each of the best-selling apparel category brands, where later after executing reviews and ratings will be automatically retrieved so that the remaining data after the cleaning process is 22.624 data. The next step is that some of the data that has been collected will be combined into one file in .csv format.After the data preparation process is complete the next step is datalabeling, for 1 to 3 stars it will be labeled bad while for 4 to 5-star reviews it will be labeled good.The data that has been labeled will go into the preprocessing process following is the preprocessing table 1.

TABLE I. PREPROCESSING RESULT

Dirty Text	Barang dsini JELEK sekali, tidak usah beliii di sini euy. Soalnya beli 3 yang 1 nya robek!!!!!!!
Case Folding	barang dsini jelek sekali, tidak usah beliii disini euy. soalnya beli 3 yang 1 nya robek!!!!!!!
Remove Punctuation & Remove Short Words	barang dsini jelek sekali tidak usah beliii disini soalnya beli yang robek
Stopword Removal	barang dsini jelek sekali tidak usah beliii disini soalnya beli robek
Word Normalizer	barang disini jelek sekali tidak usah beli disini soalnya beli robek
Stemming	barang sini jelek sekali tidak usah beli sini soal beli robek
Tokenization	“barang”, “sini”, “jelek”, “sekali”, ”tidak”, ”usah”, “beli”, “sini”, “soal”, “beli”, “robek”

In the next stage after preprocessing, the clean text data will be visualized into a wordcloud. A Word cloudis a visualization where the most frequently occurring words are large and the less frequently occurring words are smaller[23]. Can be seen in Fig.2.

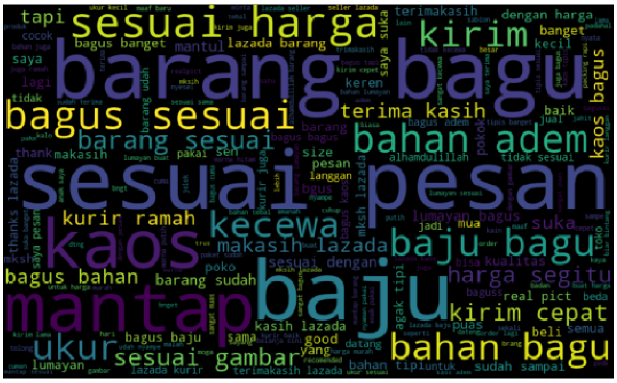

Fig. 2. Wordcloud

Based on the frequency of occurrence of the most words in Fig.2. which appears in a word cloud for 5 words with frequent occurrences, namely “barang”, ”bagus”, ”baju”, ”sesuai”, ”pesan”. The next stage is the training and evaluation of the model. The two models that have been trained using training data will be measured using a Confusion Matrix, namely accuracy, precision, recall, and f1-score which can be seen in table II

TABLE II. F1 SCORE RESULT

Dimension	Word2vec	Doc2vec
50	0.9348	0.9346
100	0.9343	0.9341
200	0.9344	0.9339
300	0.9333	0.9348

Table 3 describes the data used divided into training data and data testing, with a share of 80% for training data and 20% for data testing. Various dimensional variants will be carried out for the Word2vec and Doc2vec training. In this section, we will describe the tests carried out to obtain the best-performing model by changing the parameters used. The parameter under study is the Word2vec dimension. The measuring instrument used is the F1 score. Where the F1 score is very suitable for measuring the performance of unbalanced datasets[23]. Sizes for the dimensions Word2vec and Doc2vec used are 50, 100, 200, and 300. From the test results, it is known that Word2vec has the highest F1 Score on dimensions 50 with an F1 Score of 0.9348%, followed by vector size 200 with a value of 0.9344% while Doc2vec has the highest F1 Score on dimensions 300 with the value of F1 Score 0.9348%, followed by vector size 50 with a value of 0.9346%. The best average from the trial of the two Word2vec methods has a value of 0.9342% while Doc2vec has a value of 0.9344%.

To be clearer and can be seen in detail the increments per trial will be depicted on the graphs in Fig.2 and Fig. 3

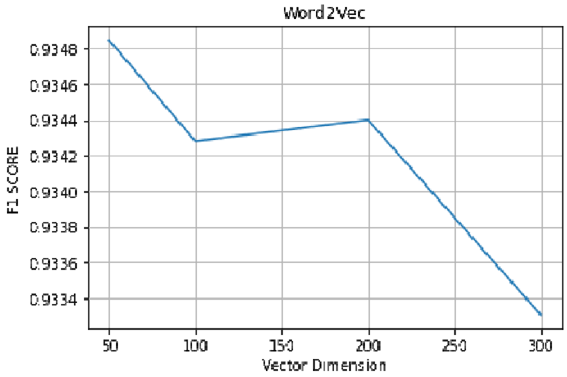

Fig. 3. Graphic Chart Word2vec

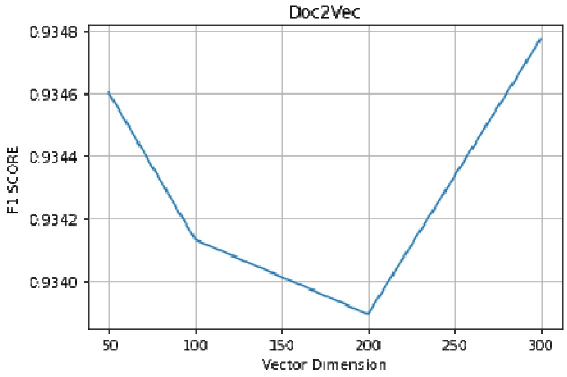

Fig. 4. Graphic Chart Doc2vec

Although for the results both Word2vec and Doc2vec have an F1 Score value of more than 0.90% in the case of this study, for trials using various vector size variants on Word2vec and Doc2vec, the results are less significant in XGBoost, this is because each method Produce a high F1-Score value not too far for the distance difference. The results show that the larger the size of the word vector dimensions is used, the training time required is also longer. Another finding that can be obtained from this research is the diversity of text data that can be processed properly through the preprocessing process. Text data is very rich in information and able to be processed properly by the word normalization stage at the preprocessing stage. This text data is processed properly and correctly, text data can have tremendous utilization potential[12]. Text processing is not an easy matter, several important steps are needed so that the processed text data can be used to find information in it. Normalized text data can go through the stemming process well, this is because abbreviated words, and non-standard words can be overcome[24]. So that it can be processed to the next process.

In this discussion, why do researchers choose two feature extraction methods?Word2vec and Doc2vec methods are used in this study because basically, Word2vec has the advantage of representing word closeness well[3], while Doc2vec is an advanced application of Word2vec[9]. For Doc2vec itself, the main advantage of the Doc2vec model is that this model represents features as dense vectors rather than conventional sparse representations which are generally able to overcome the problem of synonyms and homonyms[25]. Indeed in this case Word2vec effectively captures the semantic relationship between words. But words can only capture so much, there are times when possible there are conditions requiring a relationship between sentences and documents and not just words. For example, if there is a condition to find out whether two product reviews are duplicates of each other, it may pass the processing stage in the previous data cleaning stage

Another result that can be found in this study is to prove the assumption for a small amount of data, word2vec is not so optimal[26], so with that for the data which amounted to 22.624, the author made 1000 data with a balanced compositionand will try again. The method to be compared is word2vec, with TFIDF and Bag of words. The results of this experiment can be seen in table III.

TABLE III. F1 SCORE RESULT 1000 DATA

Model	Accuracy	Precision	Recall	F1-Score
Bag of Words	0.920	0.956	0.910	0.933
TF-IDF	0.904	0.910	0.910	0.920
Word2vec	0.825	0.861	0.849	0.855
Doc2vec	0.825	0.856	0.856	0.856

Table III. it can be seen from the experimental results of all scenarios on 4 architectural models that were tested with a composition of 1000 data, it was found that the Bag of Words and TF-IDF methods produced higher performance in terms of accuracy and f1-score compared to Word2vec and Doc2vec. Thus, the four methods used in this study have their respective advantages and disadvantages. When faced with relatively little data, the TF-IDF and Bag of Word features are better because the advantages of this traditional method assume that the fewer the frequency levels that appear, the more unique and important the word is [27], for Word2vec and Doc2vec it is more powerful when processing largedata. This is due to the small number of datasets Word2vec cannot capture the similarity of word meanings well [26].

V. CONCLUSION

There are several methods that can be used to represent text in vector form, the wrong one is by using word embedding. Word2vec in the research conducted is able to represent words well, both from 50 to 300 vector dimensions. Word2vecproduces high-performance values. Doc2vec has a better advantage over Word2vec with the condition that we will know the closeness between the review sentences, not the closeness of words anymore. Based on experiments, the performance of Word2vec and Doc2vec paired with the XGBoost Algorithm was able to classify unbalanced datasets with an average F1 Score value of 0.9342 for Word2vec and 0.9344 for Doc2vec.Word2vec and Doc2vec are more powerful when processing large data. This is due to the small number of datasets Word2vec cannot capture the similarity of word meanings well.

Further research can add a variety of datasets and can also use the multi-label method so that word embedding capabilities can be free to process text data.

ACKNOWLEDGMENT

Thank you to Universitas Amikom Yogyakarta for supporting this article can be useful.

REFERENCES

[1] D. Suleiman and A. Awajan, “Comparative Study of Word Embeddings Models and Their Usage in Arabic Language Applications,” Apr. i2019, doi: 10.1109/ACIT.2018.8672674.

[2] D. Suleiman, A. Awajan, and W. Al Etaiwi, “The Use of Hidden Markov Model in Natural ARABIC Language Processing: a survey,” *Procedia Comput. Sci.*, vol. 113, pp. 240–247, 2017, doi: https://doi.org/10.1016/j.procs.2017.08.363.

[3] A. Nurdin, B. Anggo Seno Aji, A. Bustamin, and Z. Abidin, “Perbandingan Kinerja Word Embedding Word2Vec, Glove, Dan Fasttext Pada Klasifikasi Teks,” *J. Tekno Kompak*, vol. 14, no. 2, p. 74, 2020, doi: 10.33365/jtk.v14i2.732.

[4] F. W. Kurniawan and W. Maharani, “Analisis Sentimen Twitter Bahasa Indonesia dengan Word2Vec,” vol. 7, no. 2, pp. 7821–7829, 2020.

[5] N. Purnama, “Implementasi Doc2Vec untuk rekomendasi penginapan di Bali,” *J. Tek. Inform. Unika St. Thomas*, vol. 06, no. 02, pp. 2657–1501, 2021.

[6] F. K. Khattak, S. Jeblee, C. Pou-Prom, M. Abdalla, C. Meaney, and F. Rudzicz, “A survey of word embeddings for clinical text,” *J. Biomed. Informatics X*, vol. 4, no. December, p. 100057, 2019, doi: 10.1016/j.yjbinx.2019.100057.

[7] S. Edy, R. Imam, F. Reza, and Eriszana, “SENTIMENT EMBEDDINGS DOC2VEC PADA KLASIFIKASI KELUHAN POLUSI UDARA,” vol. 9, no. 1, pp. 1–7, 2021.

[8] D. T. Hermanto, A. Setyanto, E. T. Luthfi, and U. A. Yogyakarta, “Algoritma LSTM-CNN untuk Sentimen Klasifikasi dengan Word2vec pada Media Online,” *Citec J.*, vol. Vol. 8, No, pp. 64–77, 2021.

[9] T. H. Jaya Hidayat, Y. Ruldeviyani, A. R. Aditama, G. R. Madya, A. W. Nugraha, and M. W. Adisaputra, “Sentiment analysis of twitter data related to Rinca Island development using Doc2Vec and SVM and logistic regression as classifier,” *Procedia Comput. Sci.*, vol. 197, pp. 660–667, 2022, doi: https://doi.org/10.1016/j.procs.2021.12.187.

[10] W. Christina Widyaningtyas and S. Al Faraby, “Sentiment Analysis Classification of Movie Review in English Language using Doc2Vec and Support Vector Machine (SVM),” *e-Proceeding Eng. Univ. Telkom*, vol. 5, no. 1, pp. 1570–1578, 2018.

[11] Y. Yennimar and R. Rizal, “Comparison of Machine Learning Classification Algorithms in Sentiment Analysis Product Review of North Padang Lawas Regency,” *SinkrOn*, vol. 4, p. 268, 2019, doi: 10.33395/sinkron.v4i1.10416.

[12] A. N. Rohman, R. Luviana Musyarofah, E. Utami, and S. Raharjo, “Natural Language Processing on Marketplace Product Review Sentiment Analysis,” in *2020 2nd International Conference on Cybernetics and Intelligent System (ICORIS)*, 2020, pp. 1–5. doi: 10.1109/ICORIS50180.2020.9320827.

[13] K. Afifah, I. N. Yulita, and I. Sarathan, “Sentiment Analysis on Telemedicine App Reviews using XGBoost Classifier,” in *2021 International Conference on Artificial Intelligence and Big Data Analytics*, 2021, pp. 22–27. doi: 10.1109/ICAIBDA53487.2021.9689735.

[14] S. Amien, P. Perdana, T. B. Aji, and R. Ferdiana, “Aspect Category Classification dengan Pendekatan Machine Learning Menggunakan Dataset Bahasa Indonesia (Aspect Category Classification with Machine Learning Approach Using Indonesian Language Dataset),” vol. 10, no. 3, pp. 229–235, 2021.

[15] V. Wati *et al.*, “Analisis Aspek-Aspek Kualitas Skema Database Kepegawaian Untuk Optimalisasi Perekrutan Karyawan,” *Creat. Inf. Technol. J.*, vol. 5, no. 4, p. 292, 2020, doi: 10.24076/citec.2018v5i4.194.

[16] D. Farzindar, A. A., & Inkpen, *Natural Language Processing for Social Media (G. Hirst (ed.); Third Edit)*. 2020.

[17] A. F. Hidayatullah, S. Cahyaningtyas, and A. Hakim, “Sentiment Analysis on Twitter using Neural Network: Indonesian Presidential Election 2019 Dataset,” *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1077, p. 12001, 2021, doi: 10.1088/1757-899X/1077/1/012001.

[18] L. Sihombing, H. Hannie, and B. Dermawan, “Sentimen Analisis Customer Review Produk Shopee Indonesia Menggunakan Algortima Naïve Bayes Classifier,” vol. 5, pp. 233–242, 2021, doi: 10.29408/edumatic.v5i2.4089.

[19] Q. Shuai, Y. Huang, L. Jin, and L. Pang, “Sentiment Analysis on Chinese Hotel Reviews with Doc2Vec and Classifiers,” in *2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, 2018, pp. 1171–1174. doi: 10.1109/IAEAC.2018.8577581.

[20] M. T. Akter, M. Begum, and R. Mustafa, “Bengali Sentiment Analysis of E-commerce Product Reviews using K-Nearest Neighbors,” in *2021 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD)*, 2021, pp. 40–44. doi: 10.1109/ICICT4SD50815.2021.9396910.

[21] V. Kotu and B. Deshpande, “Chapter 8 - Model Evaluation,” in *Data Science (Second Edition)*, Second Edi., V. Kotu and B. Deshpande, Eds. Morgan Kaufmann, 2019, pp. 263–279. doi: https://doi.org/10.1016/B978-0-12-814761-0.00008-3.

[22] H. Juwiantho *et al.*, “Sentiment Analysis Twitter Bahasa Indonesia Berbasis WORD2VEC Menggunakan Deep Convolutional Neural Network,” *J. Teknol. Inf. dan Ilmu Komput.*, vol. 7, no. 1, pp. 181–188, 2020, doi: 10.25126/jtiik.202071758.

[23] I. R. Hendrawan, E. Utami, and A. D. Hartanto, “Edumatic : Jurnal Pendidikan Informatika Comparison of Naïve Bayes Algorithm and XGBoost on Local Product Review Text Classification,” vol. 6, no. 1, pp. 143–149, 2022, doi: 10.29408/edumatic.v6i1.5613.

[24] B. Hakim, “ANALISA SENTIMEN DATA TEXT PREPROCESSING PADA DATA MINING DENGAN MENGGUNAKAN MACHINE LEARNING DATA TEXT PRE-PROCESSING SENTIMENT ANALYSIS IN DATA MINING USING MACHINE LEARNING School of Computer Science and Technology , Harbin Institute of Technology,” *J. Bus. Audit Inf. Syst.*, vol. 4, no. 2, pp. 16–22, 2021, doi: DOI: http://dx.doi.org/10.30813/jbase.v4i2.3000.

[25] R. Kusumaningrum, I. Nisa, R. Nawangsari, and A. Wibowo, “Sentiment analysis of Indonesian hotel reviews: from classical machine learning to deep learning,” *Int. J. Adv. Intell. Informatics*, vol. 7, p. 292, 2021, doi: 10.26555/ijain.v7i3.737.

[26] L. Efrizoni, S. Defit, M. Tajuddin, and A. Anggrawan, “Komparasi Ekstraksi Fitur dalam Klasifikasi Teks Multilabel Menggunakan Algoritma Machine Learning Comparison of Feature Extraction in Multilabel Text Classification Using Machine Learning Algorithm,” vol. 21, no. 3, 2022, doi: 10.30812/matrik.v21i3.1851.

[27] V. A. Flores and L. Jasa, “Analisis Sentimen untuk Mengetahui Kelemahan dan Kelebihan Pesaing Bisnis Rumah Makan Berdasarkan Komentar Positif dan Negatif di Instagram,” vol. 19, no. 1, 2020, doi: 10.24843/MITE.2020.v19i01.P07.