

Performance Evaluation of Classification Algorithm for Movie Review Sentiment Analysis

SUTRIAWAN^{1,2}, PULUNG NURTANTIO ANDONO¹, MULJONO¹,
 RICARDUS ANGGI PRAMUNENDAR¹

¹Department of Informatics Engineering, Universitas Dian Nuswantoro, Indonesia

²Department of Computer Science, Universitas Muhammadiyah Bima, Indonesia

Corresponding author: Muljono (e-mail: muljono@dsn.dinus.ac.id).

ABSTRACT The majority of the current research on sentiment analysis, which covers topics like political reviews, movie reviews, and product reviews, has developed quickly. The classification and clustering stage of sentiment analysis research involves a number of subjects. Some of them cover text classification comparison research and algorithm performance optimization. An intricate issue in sentiment analysis research is dealing with unstructured or semi-structured data. The sentiment analysis procedure and improving the efficacy of the classifier's algorithm are both hampered by unstructured data. In order to manage unstructured data successfully and provide accurate and relevant information, unique strategies are required. The proposed classification model performance evaluation using Support Vector Machine, Naive Bayes, K-Nearest Neighbor, and Decision Tree is specifically covered in this paper. According to the study's findings, SVM has an accuracy rate of 96% and Naive Bayes is 86%. While the decision tree's gain accuracy is 78 percent and the kNN classification model's gain accuracy is 78 percent, respectively. The test results demonstrate that SVM is superior to other classification models in terms of accuracy performance.

KEYWORDS Decision Tree; K-Nearest Neighbor; Support Vector Machine; Naive Bayes; Sentiment Analysis; Performance Evaluation.

I. INTRODUCTION

RECENT years have seen a significant expansion of sentiment analysis research, the majority of which discusses reviews of attitudes towards products, movies, politics, etc. [1]. The classification and clustering stage of sentiment analysis research involves a number of subjects. Some of them deal with text classification comparing studies and algorithm performance enhancement, both of which have recently drawn a lot of interest [2]. Artificial intelligence (AI), when used to solve problems, helps produce practical answers since it imitates human thought and behavior. Artificial intelligence (AI) in the natural language processing (NLP) domain is very related to solving sentiment analysis problems in movie reviews. In general, AI covers several scientific fields such as machine learning, natural language processing (NLP), text and speech synthesis, and text and speech synthesis [3].

Sentiment analysis is a method of natural language processing (NLP) that identifies and extracts subjective data to detect views, attitudes, and feelings about a review[4]. A film review is a piece of writing that expresses someone's thoughts on a certain movie and might elicit good or negative responses. Its purpose is to assist readers comprehend the general message of the movie so they can decide whether to watch it or not. Depending on the review, this may contribute to a movie's success or failure. Even movie makers can learn the reputation of each movie they release, learning whether a movie has a good or negative reputation through audience reviews [5]. An intricate issue in sentiment analysis research is dealing with unstructured or semi-structured data. It is difficult to manage the unstructured data since it presents a difficulty throughout the sentiment analysis process. In order to handle unstructured

data properly and produce correct information, unique strategies are required. Product reviews, particularly online film reviews, are sometimes disorganized and exceedingly challenging to understand. They typically can be found on a variety of commercial websites like Amazon, review websites like www.dpreview.com, www.imdb.com, www.cnet.com, www.zdnet.com, opinion sites like www.consumerreview.com, www.epinions.com and www.bizrate.com, as well as news or magazine feature reviews on www.rollingstone.com. The format of review opinions is typically unstructured [6]. Additionally, improving accuracy is frequently a top priority for text mining researchers, particularly in text classification. In order to increase accuracy, R. Maulana et al. modified the classification algorithm and proposed the Information Gain (IG) feature approach [7]. The same was covered by K. Lee et al. as well, which increased the reliability of predicting movie success. To achieve the best results for the categorization of film reviews, it is required to build a classification model based on these issues.

The researcher suggests a classification model text for film reviews in Indonesian utilizing machine learning and a variety of classification techniques, including Support Vector Machine, Naive Bayes, Decision Tree, and k-Nearest Neighbors. The classification algorithm is renowned for being a successful text classification technique [8]. We reevaluate the performance of the proposed classification model as we apply the classification algorithm. The goal is to determine which classification models can handle the polarity classification of the analysis of movie review with the greatest accuracy.

II. LITERATURE REVIEW

The evaluation of algorithm performance is covered in a number of research pertaining to sentiment analysis, and certain literature reviews also enhance classifier performance, e.g., A.I. Kadhim [10], V. B. Vaghela, B. M. Jadav [11], M. Karim et al. [12], Miharja, R.S. et al [13], Jagdale et al. [14].

With regards to A.I. Kadhim’s research, surveys should be conducted, and several learning methods like Bayes classifier (NB), Support vector machine (SVM), and k-Nearest Neighbors should be discussed (kNN). In this study a machine learning strategy was used to assess the effects of each technique applied to categorization of the text [9]. V. B. Vaghela, B. M. Jadav examined numerous classification methods to find the strategy that yields the most accurate classification. Using numerous classification algorithm techniques, including Support Vector Machine (SVM), Naive Bayes (NB), and Maximum Entropy, this study updates the classification of sentiment analysis (ME). This study key contribution is to provide an overview of feature selection and classification techniques that can produce outcomes with higher levels of accuracy [10]. For the classification of film review sentiments, M. Karim et al. discussed the classification of opinion text-based review sentiments and categorized them as positive or negative reviews using a variety of approaches to classification algorithms, such as Naive Bayes and Support

Vector Machine, Semantic Orientation, and SentiWordNet. The contribution of this research is presenting the performance evaluation findings of the three algorithms utilized for categorizing movie reviews and presenting adjectives and adverbs, as well as incorporating a new modified SentiWordNet Approach scheme [11]. While Miharja et al. used a machine learning strategy to compare the classification algorithm for hotel reviews, this study suggests comparing the accuracy results of the k-NN (k-Nearest Neighbor) and NB (Naive Bayes) algorithms in the dataset in the form of hotel text reviews. The best accuracy for naïve Bayes (NB) in this sample was 85.25 percent with an AUC of 0.658. Business people might use the findings of the study as a guide when deciding on their business plans [12]. RS: Jagdale et al. suggested classification techniques such Naive Bayes, Support Vector Machine, and Decision Tree to assess the sentiment of the review using the supervised learning model Decision Tree. We applied the 10-Fold Cross Validation approach to get the best categorization of the proposed model. According to this study, when the Support Vector Machine model was validated using 10-Fold Cross Validation, it had the best accuracy of 81.75 percent. This success demonstrates that SVM is the best model out of the three that were suggested [13].

III. PROPOSED METHOD

Aspect-based or object-based extraction, subjective analysis, sentiment classification, and opinion extraction are just a few of the quite sophisticated procedures that make up the text mining methodology as a whole. From filtered film review document data, subject-based research assesses the polarity classification of positive sentiment or negative sentiment [14]. In order to categorize sentiment film reviews, this study uses a machine learning model based on supervised learning. We suggest Support Vector Machine (SVM), Naive Bayes (NB), Decision Tree (DT), and k-Nearest Neighbors as categorization models (kNN). The four methods are well-known for handling text classification models quite effectively and are quite popular.

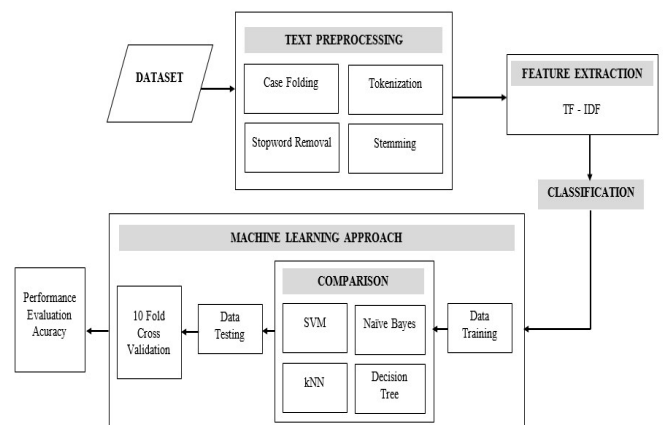


Figure 1. Proposed Method

The framework of the proposed research method consists of five (5) processes that must be completed, including the dataset

entry stage, preprocessing, feature extraction, classification using a machine learning approach by implementing the four classification models shown in Fig. 1, and the proposed model evaluation stage. The normalization of the dataset into vector format, which will be used as the input for the classification method, and which is composed of case folding, tokenization, stopword removal, and stemming, is a component of the preprocessing stage. Two (2) data sets, namely training data and testing data, can be separated and partitioned for feature extraction by employing TF-IDF to generate features from film review data. Then, employing each algorithm Support Vector Machine, Naive Bayes, K-Nearest Neighbor, and Decision Tree the categorization is carried out. We obtain 90% training data and 10% test data using 10-fold cross validation. In order to gauge the effectiveness of the algorithm, classification is performed and assessed using the confusion matrix technique.

A. DATASET COLLECTION

The dataset used in this study for movie reviews was compiled from the work of Y. Nurdiansyah et al. [15]. This dataset consists of information from Indonesian movie reviews. There are two separate datasets totaling 500 data records: a positive dataset with 250 documents and a negative dataset with 250 records. Using datasets labeled with positive and negative classifications, this dataset is used to test the effectiveness of the Support Vector Machine classification algorithm. It is divided into two types: training data and film review test data. We have built a sentiment analysis model using 90% of the training dataset and 10% of the test dataset. Table 1 displays an illustration of the dataset we made use of.

Table 1. Dataset of film reviews in Indonesian

Doc. ID	Movie Review	Sentiment
D1	<i>Nothing as unique or strange as the title might imply only another heartbreaking melodrama.....</i>	Negative
D2	<i>What happened over the past 20 minutes is inexcusable in my eyes. Yes, it was a truly outrageous joke....</i>	Negative
D3	<i>Shimizu mulai unjuk gigi Ia membawa teror horor ala asia khususnya Jhorror ke dalam kabin pesawat komersial Amerika....</i>	Positive
....
D499	<i>Shimizu begins to brag. He infiltrates the cabin of an American commercial aircraft with Asian-style horror dread, especially Jhorror.....</i>	Negative
D500	<i>The spectator is only given uninteresting and tiresome situations, many of which appear excessive, for almost the first thirty minutes.....</i>	Negative

B. TEXT PREPROCESSING

To enhance classifier performance, preprocessing is crucial. We use a number of preprocessing techniques at this point to remove noise from the data. Case folding, which standardizes letters, is a data preprocessing method used in natural language processing [16]. Text data can be divided into many tokens via tokenization. The spacing, spelling, and other aspects of the text are distinctive. In order to decrease textual input and enhance the performance of the classification algorithm, stopword elimination is particularly crucial in sentiment analysis when examining words for conjunctions, prepositions, pronouns, and unrelated words [17]. And stemming is used to change terms into primary words. Additionally, stemming is employed to convert terms into their basic word forms or get rid of affixes like me-, -kan, -ter, -nya, etc. Stemming aids in normalizing the data, making it a necessary word to streamline the subsequent process [18].

C. FEATURE EXTRACTION

The TF-IDF weighting method transforms data that has successfully completed the preprocessing stage into a numeric form. Inverse frequency term of documents. The weights between words that frequently appear can be balanced with the use of the TF-IDF. The relevance of a term in a document is therefore shown by the TF-IDF value [19]. By assigning a weight to each phrase in the document, TF-IDF calculates how closely related the terms are to one another. The frequency of a word occurring in a document and the reverse frequency of the document containing the expression are combined in the TF-IDF. To start, be aware that the TF value with each word's weight is 1 [20]. In the interim, the IDF value can be represented as:

$$TF - IDF_{ij} = tf_{ij} * \log \left(\frac{N}{df_{i+1}} \right). \quad (1)$$

Each word in the corpus is counted by IDF based on the total number of documents and the frequency of each individual document (i.e., the word with the value). The TF-IDF approach is employed to hasten the term calculating procedure. In addition to expediting term calculations, TF-IDF has reliable results and can carry out effective weighing. The words in the paper that translate to numbers are the features. Textual input is translated to numeric using a count vectorizer approach. The count vectorizer is a matrix where each row stands in for a word and each column for a feature. When choosing training sets and analyzing text categorization, this matrix can be input for classification algorithms and validation approaches [19].

D. SENTIMENT CLASSIFICATION APPROACH

At this point, we use the Support Vector Machine, Naive Bayes, K-Nearest Neighbor, and Decision Tree classification method approaches. In order to assess the effectiveness of the suggested classification model, the classifier seeks to ascertain the level of sentiment obtained, namely positive and negative.

An efficient machine learning technique for the study of text classification is naive Bayes. If the total number of documents falls into category k , then applying each document to class $c = \arg \max c$ is one method of text classification. The Bayes rule can be used to determine the likelihood that a document is in the class.

$$P(c|d) = \frac{p(c)P(d/c)}{p(d)}, \tag{2}$$

$$C_{MAP} = \arg \max P(d|c)P(c), \tag{3}$$

$$= \arg \max P(x_1, x_2 \dots, x_n|c)p(c), \tag{4}$$

Where, $P(d)$ does not play a role in choosing c to estimate the term $P(c|d)$. Naive Bayes outlines assuming f_i In the document occurs independently given class d . Then the equation becomes:

$$P(c|d) = \frac{P(c) (\prod_{i=1}^m P(f_i | c)n_i(d))}{P(d)}. \tag{5}$$

The training method consists of estimating the relative frequencies of $P(c)$ and $P(f_i | c)$ using add-one smoothing [22, 23].

K-Nearest Neighbor is a classification technique that chooses a number of labeled training instances that are the

closest to each other. This approach is effective and can even manage categorization for documents with several categories. Simply put, when given a lot of training samples, the KNN method takes longer to identify objects [24]. The class with the most complimentary memberships is classified by KNN. Objects are categorized by a majority vote of their neighbors, with the k closest neighbors' most prevalent class being assigned to each object [25]. An algorithm known as a decision tree can create a detailed training model for true-false questions in a tree form, such as a flowchart [26]. Each internal node of the decision tree, which is a classification method, represents a test on an attribute. Each leaf node represents a prediction class, and each branch represents a result test [9]. By processing requests, starting at the root and continuing until it reaches a certain leaf that represents the document classification objectives, the decision tree may categorize documents [27].

IV. RESULTS AND ANALYSIS

A. PERFORMANCE EVALUATION ANALYSIS

Combinations of algorithms are available for text classification through text mining. In order to understand how classification algorithms, feature extraction, and other algorithms were applied in earlier studies, we did a comparison study with various different types of algorithms. A comparison of the findings from earlier investigations is shown in Table 2.

Table 2. Research Performance Evaluation Comparison

No	Author name & title	Dataset	Feature	Performance measure	Algorithm	Accuracy
1	A. I. Kadhim. Survey on supervised machine learning techniques for automatic text classification [10].	Movie review dataset V2.0 and Movie review dataset V1.0	TF-IDF	Recall, Precision, Accuracy	NB	70.9%
					kNN	70.5%
					K*	69.4%
					SVM	76%
					DT-J48	69.9%
2	V. B. Vaghela, B. M. Jadav. Analysis of Various Sentiment Classification Techniques, [11].	Movie Review	Count Vector, TF-IDF, Information Gain, Mutual information, Chi-square	Recall, Precision, Accuracy	SVM	82.9 %
					Naïve Bayes	81 %
					Maximum Entropy	81.5%
3	M. Karim et al. Sentiment Analysis of Textual Reviews [12].	Movie Review	TF-IDF	Accuracy, precision, Recall, F-measure.	Naive Bayes	89.4%
					SVM	83.3%
4	Miharja, et al. Comparison Of Machine Learning Classification Algorithm On Hotel Review Sentiment Analysis [13].	Hotel Review	-	Accuracy, AUC	Naive Bayes	85.25%
					kNN	60.50%
5	Jagdale et al. Sentiment Analysis of Customer Product Reviews Using Machine Learning [14].	E-commerce Giant Amazon.com.	-	Accuracy.	Naïve Bayes	66.95%
					SVM	81.77%
					Decision Tree	74.75%

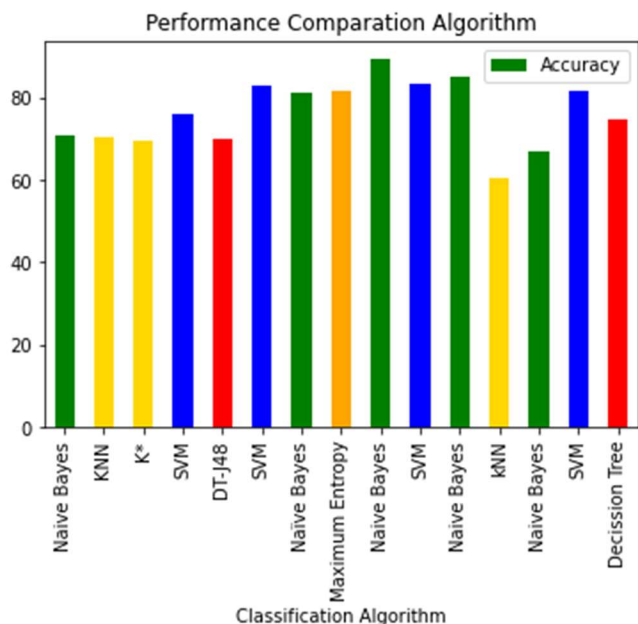


Figure 2. Performance Evaluation Comparison Algorithm.

The performance of the classification algorithm is compared to other classification models put out in earlier works, as did A.I. Kadhim, in Table 2 and Fig. 2. They contrasted the Naive Bayes (NB), kNN, K*, SVM, and DT-J48 classification methods. His research findings indicated that the SVM algorithm had the best accuracy of 76% [9]. SVM, Naive Bayes (NB), and Maximum Entropy classification models were all evaluated by V. B. Vaghela, B. M. Jadav. The findings revealed that SVM had the highest accuracy, at 82.9% [10]. A Naive Bayes (NB) classification model, SVM, was proposed by M. Karim et al. The test results indicated that the best accuracy was 89.4% [11]. Naive Bayes (NB) and K-Nearest Neighbor were proposed by Miharja et al. to compare the classification algorithms using the Hotel Review dataset (kNN). The results of these experiments demonstrated an accuracy of 60.50 percent with kNN and 85.25 percent with Naive Bayes [12]. Jagdale et al. compared the classification algorithms Naive Bayes (NB), SVM, and Decision Tree in their discussion of the same topic (DT). The support vector machine (SVM) algorithm test yielded the best accuracy results, with a score of 81.77 percent [13].

We carried out experiments to test and assess the performance models of the classifiers that we have presented, based on the findings of the analysis of the performance of the classification algorithms proposed in prior works. The Support Vector Machine (SVM), Naive Bayes (NB), K-Nearest Neighbor (kNN), and Decision Tree are the four (four) machine learning models we utilized, as shown in Fig. 1. (DT). To categorize the sentiment of movie reviews into positive and unfavorable viewpoints, we train a machine learning model.

B. EXPERIMENT AND RESULTS USING 10-FOLD CROSS VALIDATION

At this point, we apply 10-Fold Cross Validation to the TF-IDF weighted extracted feature set for the complete film review document data set to assess the efficacy of the classification algorithm. This analysis tries to identify the algorithm optimal performance for categorizing positive and negative feelings from movie reviews. Table 3 displays the trial outcomes utilizing the four (4) classifier model techniques.

Table 3. Performance Comparison using 10-Fold Cross Validation

10-Fold Cross Validation	Classification Algorithm			
	Support Vector Machine (SVM)	Naive Bayes (NB)	K-Nearest Neighbor (kNN)	Decision Tree (DT)
1	93,3%	95,5%	62,2%	77,7%
2	88,8%	88,8%	60,%	77,7%
3	86,6%	84,4%	68,8%	80%
4	86,6%	91,1%	64,4%	73,3%
5	91,1%	91,1%	66,6%	75,5%
6	88,8%	88,8%	68,8%	71,1%
7	86,6%	84,4%	64,4%	73,3%
8	88,8%	95,5%	75,5%	80%
9	91,1%	95,5%	68,8%	75,5%
10	91,1%	91,1%	73,3%	84,4%
Average	89,28%	90,62%	67,28%	76,85%

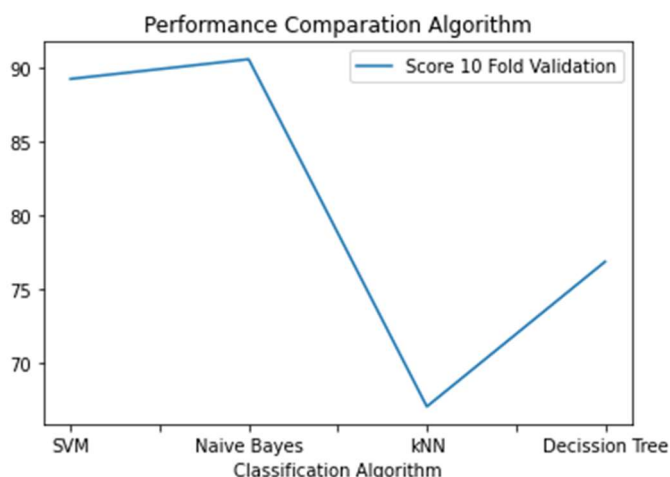


Figure 3. Cross Validation Performance Comparison

In accordance with Table 3 and Fig. 3 SVM scores 89.28 percent on average, Naive Bayes scores 90.62 percent, kNN scores 67.28 percent, and the decision tree scores 76.85 percent in the test utilizing 10-Fold Cross Validation. Naive Bayes produces a high-performance value when compared to other classification models, according to the average value produced from this classification algorithm performance. We use the Confusion Matrix Method to gauge how well the suggested

categorization model performs. The number of each data point in a specific class is shown in the matrix in each column. The number of each piece of information in the prediction class is represented by a row.

Table 4. Confusion Matrix

Class	Positive	Negative
Positive	TP (True Positive)	FP (False Positive)
Negative	FN (False Negative)	TN (True Negative)

$$Precision = \frac{TP}{TP + FP} \tag{6}$$

$$Recall = \frac{TP}{TP + FN} \tag{7}$$

$$F1 - score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{8}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{9}$$

Accuracy, precision, recall, and F1-score are some examples of performance classifications that may be calculated very effectively using the confusion matrix approach. The accuracy value used to gauge classification performance based on the confusion matrix can be validated by prediction results [28]. The confusion matrix value is displayed in Table 5.

Table 5. Values of Confusion Matrix

Class Prediction	True Positive (TP)	True Negative (TN)	False Positive (FP)	False Negative (FN)
Support Vector Machine (SVM)	20	28	2	0
Naïve Bayes (NB)	21	22	3	4
K-Nearest Neighbor	17	22	8	3
Decision Tree (DT)	17	22	8	3

Table 6. Comparison of accuracy algorithm

Classification Algorithm	Accuracy
Naive Bayes (NB)	86%
Support Vector Machine (SVM)	96%
K-Nearest Neighbor (KNN)	78%
Decision Tree (DT)	78%

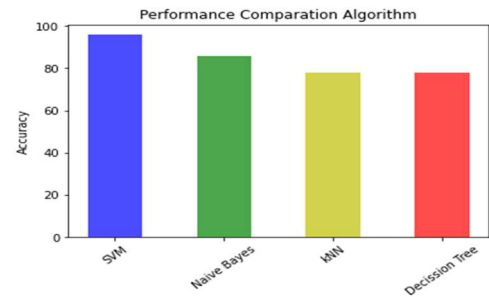


Figure 4. Accuracy algorithm comparison

The best accuracy achieved with SVM is 96%, and Naive Bayes is 86%, according to Table 6. The classification results utilizing the Support Vector Machine (SVM) algorithm show the highest accuracy value when compared to other classification models, with accuracy values of 80% for the kNN classification model and 80% for the decision tree.

V. CONCLUSION

In this study, we discussed the benefits and drawbacks of each classification model. The outcomes of our experiments demonstrate that SVM performs better in terms of accuracy when compared to alternative classification models. In addition, Naive Bayes is chosen because of its performance-enhancing low memory and processing needs. Even though decision trees take a long time to process during training, they can perform well. Similar to this, the kNN classification model performs well when used to categorize sentiment analysis video polarity. We calculated the accuracy, precision, recall, and F1-score values in order to assess the classification algorithm performance. For the classification of positive and negative polarity attitudes from Indonesian film reviews, we compared the performance of the classification methods Naive Bayes, Support Vector Machine (SVM), K-Nearest Neighbor (kNN), and Decision Tree (DT). The results of the test show that the SVM algorithm variety produces the greatest outcomes, with a 96 percent accuracy rate. SVM performs better than other classification algorithm approaches as a result.

References

- [1] S. Widya Sihwi, I. Prasetya Jati, and R. Anggrainingsih, "Twitter sentiment analysis of movie reviews using information gain and Naïve Bayes classifier," *Proceedings of the 2018 Int. Semin. Appl. Technol. Inf. Commun. Creat. Technol. Hum. Life, iSemantic 2018*, pp. 190–195, 2018. <https://doi.org/10.1109/ISEMANTIC.2018.8549757>.
- [2] B. Das and S. Chakraborty, "An improved text sentiment classification model using TF-IDF and next word negation," 2018, [Online]. Available at: <http://arxiv.org/abs/1806.06407>
- [3] R. I. Mukhamediev et al., "Review of artificial intelligence and machine learning technologies: Classification, restrictions, opportunities and challenges," *Mathematics*, vol. 10, no. 15, pp. 1–25, 2022. <https://doi.org/10.3390/math10152552>.

- [4] P. Kalaivani and K. L. Shunmuganathan, "An improved K-nearest-neighbor algorithm using genetic algorithm for sentiment classification," *Proceedings of the 2014 Int. Conf. Circuits, Power Comput. Technol. ICCPCT 2014*, pp. 1647–1651, 2014. <https://doi.org/10.1109/ICCPCT.2014.7054826>.
- [5] T. T. Thet, J.-C. Na, and C. S. G. Khoo, "Aspect-based sentiment analysis of movie reviews on discussion boards," *J. Inf. Sci.*, vol. 36, no. 6, pp. 823–848, 2010. <https://doi.org/10.1177/0165551510388123>.
- [6] P. Chaovalit and L. Thou, "Movie review mining: A comparison between supervised and unsupervised classification approaches," *Proc. Annu. Hawaii Int. Conf. Syst. Sci.*, vol. 00, no. C, p. 112, 2005. <https://doi.org/10.1109/HICSS.2005.445>.
- [7] R. Maulana, P. A. Rahayuningsih, W. Irmayani, D. Saputra, and W. E. Jayanti, "Improved accuracy of sentiment analysis movie review using support vector machine based information gain," *J. Phys. Conf. Ser.*, vol. 1641, no. 1, 2020. <https://doi.org/10.1088/1742-6596/1641/1/012060>.
- [8] K. Lee, J. Park, I. Kim, and Y. Choi, "Predicting movie success with machine learning techniques: ways to improve accuracy," *Inf. Syst. Front.*, vol. 20, no. 3, pp. 577–588, 2018. <https://doi.org/10.1007/s10796-016-9689-z>.
- [9] A. H. Aliwy and E. H. A. Ameer, "Comparative study of five text classification algorithms with their improvements," *Int. J. Appl. Eng. Res.*, vol. 12, no. 14, pp. 4309–4319, 2017. <https://files2.shewayaya.com/files/62055.pdf>.
- [10] A. I. Kadhim, "Survey on supervised machine learning techniques for automatic text classification," *Artif. Intell. Rev.*, vol. 52, no. 1, pp. 273–292, 2019. <https://doi.org/10.1007/s10462-018-09677-1>.
- [11] V. B. Vaghela, B. M. Jadav, "Analysis of various sentiment classification techniques," *Int. J. Comput. Appl.*, vol. 140, no. 3, pp. 22–27, 2016. <https://doi.org/10.5120/ijca2016909259>.
- [12] M. Karim and S. Das, "Sentiment analysis on textual reviews," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 396, no. 1, pp. 122–127, 2018. <https://doi.org/10.1088/1757-899X/396/1/012020>.
- [13] J. Miharja, J. L. Putra, and N. Hadianto, "Comparison of machine learning classification algorithm on hotel review sentiment analysis (Case study: Luminor hotel Pecenongan)," *J. Pilar Nusa Mandiri*, vol. 16, no. 1, pp. 59–64, 2020. <https://doi.org/10.33480/pilar.v16i1.1131>.
- [14] R. S. Jagdale, V. S. Shirsat, and S. N. Deshmukh, "Sentiment analysis on product reviews using machine learning techniques," *Adv. Intell. Syst. Comput.*, vol. 768, pp. 639–647, 2019. https://doi.org/10.1007/978-981-13-0617-4_61.
- [15] B. Saberi and S. Saad, "Sentiment analysis or opinion mining: A review," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 7, no. 5, pp. 1660–1666, 2017. <https://doi.org/10.18517/ijaseit.7.5.2137>.
- [16] Y. Nurdiansyah, S. Bukhori, and R. Hidayat, "Sentiment analysis system for movie review in Bahasa Indonesia using Naive Bayes classifier method," *J. Phys. Conf. Ser.*, vol. 1008, no. 1, 2018. <https://doi.org/10.1088/1742-6596/1008/1/012011>.
- [17] R. I. Pristiyanti, M. A. Fauzi, and L. Muflikhah, "Sentiment analysis summarizing film reviews using information gain and K-nearest neighbor methods," *J. Pengemb. Teknol. Inf. dan Ilmu Komput. Univ. Brawijaya*, vol. 2, no. 3, pp. 1179–1186, 2018. (in Indonesian), [Online]. Available at: <http://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/1140>.
- [18] R. Hidayat and S. Minati, "Comparative Analysis of Text Mining Classification Algorithms for English and Indonesian Qur'an Translation," *IJID (International J. Informatics Dev.)*, vol. 8, no. 1, p. 47, 2019. <https://doi.org/10.14421/ijid.2019.08108>.
- [19] B. V. Indriyono, E. Utami, and A. Sunyoto, "Utilization of the Porter Stemmer Algorithm for Indonesian in the process of classifying book types," *J. Buana Inform.*, vol. 6, no. 4, pp. 301–310, 2015. (in Indonesian) <https://doi.org/10.24002/jbi.v6i4.462>.
- [20] A. Tripathy, A. Agrawal, and S. K. Rath, "Classification of sentimental reviews using machine learning techniques," *Procedia Comput. Sci.*, vol. 57, no. March, pp. 821–829, 2015. <https://doi.org/10.1016/j.procs.2015.07.523>.
- [21] D. Kim, D. Seo, S. Cho, and P. Kang, "Multi-co-training for document classification using various document representations: TF-IDF, LDA, and Doc2Vec," *Inf. Sci. (Njy)*, vol. 477, pp. 15–29, 2019. <https://doi.org/10.1016/j.ins.2018.10.006>.
- [22] P. Dellia and A. Tjahyanto, "Tax complaints classification on twitter using text mining," *IPTEK J. Sci.*, vol. 2, no. 1, p. 11, 2017. <https://doi.org/10.12962/j23378530.v2i1.a2254>.
- [23] C. J. Huang, Y. J. Yang, D. X. Yang, and Y. J. Chen, "Frog classification using machine learning techniques," *Expert Syst. Appl.*, vol. 36, no. 2 PART 2, pp. 3737–3743, 2009. <https://doi.org/10.1016/j.eswa.2008.02.059>.
- [24] K. Kowsari, K. J. Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown, "Text classification algorithms: A survey," *Inf.*, vol. 10, no. 4, pp. 1–68, 2019. <https://doi.org/10.3390/info10040150>.
- [25] P. Y. Pawar and S. H. Gawande, "A comparative study on different types of approaches to text categorization," *Int. J. Mach. Learn. Comput.*, vol. 2, no. 4, pp. 423–426, 2012. <https://doi.org/10.7763/IJMLC.2012.V2.158>.
- [26] M. Mamtesh and S. Mehla, "Sentiment analysis of movie reviews using machine learning classifiers," *Int. J. Comput. Appl.*, vol. 182, no. 50, pp. 25–28, 2019. <https://doi.org/10.5120/ijca2019918756>.
- [27] V. C. Pande and A. S. Khandelwal, "Comparative analysis of various text classification algorithms," *Int. J. Comput. Sci. Inf. Technol.*, vol. 8, no. 5, pp. 574–577, 2017.
- [28] V. Korde, "Text classification and classifiers: A survey," *Int. J. Artif. Intell. Appl.*, vol. 3, no. 2, pp. 85–99, 2012. <https://doi.org/10.5121/ijai.2012.3208>.
- [29] R. B. Purnomoputra and U. N. Wisesty, "Sentiment analysis of movie reviews using Naïve Bayes method with Gini index feature selection," no. July, pp. 85–94, 2019. <https://doi.org/10.34818/jdsa.2019.2.36>.



SUTRIAWAN, earned his bachelor's degree in informatics engineering with a focus in 2018 from Ahmad Dahlan University in Yogyakarta, Indonesia's Daerah Istimewa Yogyakarta (DIY). He finished his undergraduate studies in 2018 and graduated from Dian Nuswantoro University in Indonesia with a master's in informatics engineering. His work focuses on text mining and machine learning are related to sentiment analysis.



Pulung Nurtantlo Andono is currently Associate Professor Informatics Engineering. He received master of computer science from Universitas Dian Nuswantoro, Semarang, Indonesia and doctoral degree in Computer Science from Institut Teknologi Sepuluh Nopember (ITS) Surabaya. He is currently a lecturer at Universitas Dian Nuswantoro, Semarang, Indonesia. His research interests.



Muljono is with Informatics Engineering Department, Dian Nuswantoro University, Semarang, Jawa Tengah, Indonesia. He received a bachelor's degree in Mathematics Department at Diponegoro University, Semarang in 1996, a master's degree in Informatics Engineering, STTIBI, Jakarta in 2001, and a Ph.D. degree at Electrical Engineering Department, Institut Teknologi Sepuluh Nopember (ITS), Surabaya in 2016, Indonesia. He joined an Internship Program at the

School of Media Science, Tokyo University of Technology Japan, in 2014. His research interests are speech processing, artificial intelligence, and natural language processing.



Ricardus Anggi Pramunendar, has completed his undergraduate studies (S1), majoring in informatics engineering in the Department of Computer Science, Dian Nuswantoro University, Semarang, Jawa Tengah Indonesia. Then he continued his studies at the Master of computer science program at the Technical University Malaysia Malacca (UTEM), Malaysia. He has completed the Computer Science Ph.D. in electrical engineering and information technology, Universitas Gadjah Mada. Yogyakarta, Daerah Istimewia Yogyakarta (DIY), Indonesia.

...