



OPEN ACCESS

Vietnam Journal of Computer Science

Vol. 10, No. 4 (2023) 485–498

© The Author(s)

DOI: [10.1142/S2196888823500100](https://doi.org/10.1142/S2196888823500100)



World Scientific

www.worldscientific.com

Multinomial Naïve Bayes Classifier for Sentiment Analysis of Internet Movie Database

Christine Dewi*

Faculty of Information Technology

Satya Wacana Christian University, Salatiga, Indonesia

christine.dewi@uksw.edu

Rung-Ching Chen

Department of Information Management

Chaoyang University of Technology

Taichung, Taiwan

crching@cyut.edu.tw

Henoch Juli Christanto

Department of Information System

Atma Jaya Catholic University of Indonesia

Jakarta 12930, Indonesia

henoch.christanto@atmajaya.ac.id

Francesco Cauteruccio

Department of Information Engineering

Polytechnic University of Marche

Ancona 60121, Italy

f.cauteruccio@univpm.it

Received 10 June 2023

Revised 4 July 2023

Accepted 15 July 2023

Published 9 August 2023

Sentiment analysis (SA), also known as opinion mining, is a natural language processing (NLP) technique used to determine the sentiment or emotional tone behind a piece of text. It involves analyzing the text to identify whether it expresses a positive, negative, or neutral sentiment. SA can be applied to various types of text data such as social media posts, customer reviews, news articles, and more. This experiment is based on the Internet Movie Database (IMDB) dataset, which comprises movie reviews and the positive or negative labels related to them. Our research experiment's objective is to identify the model with the best accuracy and the most generality.

*Corresponding author.

This is an Open Access article published by World Scientific Publishing Company. It is distributed under the terms of the Creative Commons Attribution 4.0 (CC BY) License which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

Text preprocessing is the first and most critical phase in an NLP system since it significantly impacts the overall accuracy of the classification algorithms. The experiment implements unsupervised sentiment classification algorithms including Valence Aware Dictionary and sentiment Reasoner (VADER) and TextBlob. We also examine the supervised sentiment classifications methods such as Naïve Bayes (Bernoulli NB and Multinomial NB). The Term Frequency-Inverse Document Frequency (TFIDF) model is used to feature selection and extractions. The combination of Multinomial NB and TFIDF achieves the highest accuracy, 87.63%, for both classification reports based on our experiment result.

Keywords: Sentiment analysis; Naïve Bayes; Bernoulli NB; multinomial NB; TFIDF; sentiment classifications.

1. Introduction

The exponential growth of digital information has resulted in the rapid development of a new field of research: Sentiment analysis (SA). In the modern science of artificial intelligence (AI), SA is a crucial technique for gleaning emotional information from massive volumes of data. SA is also known as “sentiment analysis.”¹ SA^{2,3} has been improved over the years using several machine learning and dictionary-based algorithms in order to improve its accuracy. Prior knowledge also plays an important role in conveying the polarity of views well with the advent of deep learning algorithms⁴ in SA.

SA can be applied in various types of text data such as social media posts, customer reviews, news articles, and more. It is widely used in a range of applications, including: (1) Brand monitoring: Companies use SA to monitor online conversations about their brand and products, helping them understand customer opinions and sentiment towards their offerings.^{5,6} (2) Customer feedback analysis: SA enables businesses to analyze customer feedback from surveys, reviews, and social media to gain insights into customer satisfaction and identify areas for improvement.^{7,8} (3) Market research: SA can be employed to analyze public sentiment towards specific products, services, or market trends, providing valuable insights for market research and decision-making.^{9,10} (4) Social media analysis: By analyzing sentiment in social media posts, companies and organizations can gauge public opinion on various topics, track trends, and understand the impact of their social media campaigns.^{11,12} (5) Reputation management: SA helps organizations monitor their online reputation by identifying negative sentiment or potential PR issues, allowing them to address problems promptly and maintain a positive image. (6) Political analysis: SA is used to gauge public opinion towards political candidates, parties, or specific policies, assisting in election campaigns and policy development.^{13,14}

In the industrial context, businesses primarily use SA to collect and assess client feedback. The fields of natural language processing (NLP) and SA are inextricably linked.¹⁵ The majority of the material on the internet comes in the form of natural language, which robots cannot understand owing to its complexity and inter-word semantics.¹⁶ NLP is analyzing these natural texts to create things that the machine can understand for SA. The Internet Movie Database (IMDB) dataset comprises of 50,000 reviews, equally split between 25,000 train and 25,000 test reviews. Positive or negative movie reviews are categorized, and the task is to guess the sentiment of an

unseen review. The sentiment of a movie review is usually associated with a different rating, which can be used for classification dilemmas. It can be used as a reference instrument for movie preference.^{17,18}

This study is an expanded version of the article presented at the ACIIDS 2022 conference.¹⁹ In that article, only Linear Model and Naïve Bayes are proposed for the sentiment classifications. The following are the significant contributions of this work: (1) The study is based on the IMDB dataset, which comprises movie reviews and the positive or negative labels that relate to them. (2) The goal of our study experiment is to find the model with the highest accuracy and the greatest generality. (3) The following classifiers are used in this work: Unsupervised Learning (VADER and Text Blob), and Supervised Learning (Naïve Bayes including the Bernoulli NB and Multinomial NB). Different techniques, including Count Vectorizer, Term Frequency-Inverse Document Frequency (TFIDF) model Vectorizer, minimum–maximum number of words, and max features, are implemented. We can observe that the Multinomial Naïve Bayes model performs well compared to other methods. The rest of the paper is structured in the following manner. Section 2 introduces the related works. Section 3 describes the research process. Section 4 discusses the experiment's findings and conclusions. Section 5 summarizes the conclusion and future research endeavors.

2. Related Work

2.1. Sentiment analysis (SA)

Researchers have been working on various recommendation algorithms based on text data supplied by internet users over the last couple of years. Zirn *et al.*²⁰ developed a completely automated system for fine-grained SA at the sub-sentence level, incorporating several sentiment lexicons, neighborhood links, and discourse linkages. Appel *et al.*²¹ established a hybrid strategy based on ambiguity management, semantic rules, and a sentiment lexicon using Twitter sentiment and movie review datasets. The authors evaluated the performance of their suggested hybrid system to that of conventional supervised algorithms such as Naïve Bayes (NB) and Maximum Entropy (ME). The suggested approach outperforms supervised methods in terms of precision and accuracy. Similarly, Pang *et al.*²² used unigram, bigram, and unigram models to analyze IMDB movie review data SA. They classified the data into two classes using NB, ME, and SVM classifiers.²³ SA has improved over the years using several dictionary-based and machine learning algorithms to improve accuracy. Prior knowledge has also played a significant role in properly conveying the polarity of views with the emergence of deep learning algorithms in SA.

2.2. Naïve Bayes classifier

The NB classifier is a simple but effective probabilistic algorithm used for classification tasks, including SA. It's based on Bayes' theorem, which describes the

probability of an event given prior knowledge or evidence.²⁴ The “Naïve” assumption in NB refers to the assumption whose features are conditionally independent, meaning that the presence or absence of one feature does not affect the presence or absence of other features. NB is a well-known classification technique in data mining.²⁵ The NB classification model computes a class’s posterior probability based on its posterior probability distribution using the word distribution in the text. The model is based on bag-of-words (BOWs) feature extraction, which does not consider the location of the word in the text. It makes use of the Bayes Theorem to forecast the likelihood that a given feature set belongs to a certain label in a given situation.²⁶

$$P(\text{label}|\text{features}) = \frac{P(\text{label}) * P(\text{features}|\text{label})}{P(\text{features})} \quad (1)$$

where $P(\text{label}|\text{features})$ quantifies the likelihood that a feature set is associated with a given label. $P(\text{label})$ is the label’s previous estimate. $P(\text{label}|\text{features})$ denotes the likelihood that the provided feature set is associated with this label, and $P(\text{features})$ is the previous estimate for the occurrence of this feature collection. However, this categorization technique involves a basic assumption, namely that words in a review, category pair appear independently of other terms.

When dealing with discrete values, such as word counts, NB classification is the best option. Further, we expect it to show the best accuracy and count the occurrences of a word in the complement to the class. The classification result is the class with the lowest sum of weights for each word in the message, which is represented by the class with the lowest value.

Bernoulli NB is a variant of the NB classifier that is specifically designed for binary feature data, where each feature can take on one of two values (typically 0 or 1).^{27,28} It is commonly used in text classification tasks, including SA. The Bernoulli NB classifier assumes that each feature is conditionally independent of others, given the class label, and follows a Bernoulli distribution.²⁹ It calculates the probability of a document belonging to a particular class (e.g. positive, or negative sentiment) based on the presence or absence of binary features. The Bernoulli formula is very similar to the multinomial formula, with the exception that the input is a set of Boolean values (whether the word is in the message or not), rather than a set of frequencies. Consequently, the algorithm explicitly penalizes the absence of a feature (word in the message is absent from the vocabulary), whereas the multinomial approach makes use of the smoothing parameter for the values that are not present in the algorithm.^{30,31}

Lee *et al.*³² developed a novel approach for determining feature weights in Naïve with accent Bayesian learning by averaging the Kullback–Leibler measure across feature values. Additionally, they proposed a novel weight assignment paradigm for categorization learning, nicknamed the value weighting approach. Rather than weighing each attribute, they assign a different weight to each value. In Ref. 33, the authors present a weighting attributes to Alleviate NB’ Independence Assumption (WANBIA), that optimizes the negative conditional log-likelihood or the mean

squared error objective functions. Their experiment conduct rigorous analyses and demonstrate that WANBIA is a viable alternative to state-of-the-art classifiers such as Random Forest and Logistic Regression.^{31,34}

Multinomial NB is another variant of the NB classifier that is commonly used in text classification tasks, including SA. It is specifically suited for features that represent word frequencies or counts, which are typically obtained using techniques like the bag-of-words model or TF-IDF.^{35,36} The Multinomial NB classifier assumes that the features are conditionally independent of each other, given the class label, and follows a multinomial distribution. It calculates the probability of a document belonging to a particular class based on the frequencies or counts of features (words) in the document.

2.3. Performance evaluation metrics

Our experiments used conventional performance indicators to evaluate our model's performance on IMDB datasets, notably the F1 and accuracy scores and their related class support divisions. Precision and recall are defined in Formulas (2) and (3).³⁷ Moreover, accuracy and F1 are defined in Formulas (4) and (5).^{38,39}

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad (2)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (3)$$

$$\text{Accuracy score} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (4)$$

$$\text{F1} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (5)$$

where True Positive (TP) is the number of reviews sorted properly into the appropriate sentiment classifications. Next, False Positive (FP) is the number of reviews assigned to an emotion class to which they do not belong. Hence, False Negative (FN) is the number of reviews labeled as not belonging to a sentiment category in which they really fit.⁴⁰

3. Methodology

3.1. Research workflow

This section will describe our research workflow, as shown in Fig. 1. Furthermore, the IMDB dataset is gathered and imported. Pre-processing is applied to the data to eliminate noise and tidy it up before further processing. Next, word embedding is a technique for expressing words in a low-dimensional space, most typically represented as real-valued vectors. It enables words with similar meanings and semantics to be expressed more closely together than words with less comparable meanings and semantics to be rendered farther apart. Therefore, the experiment implements the

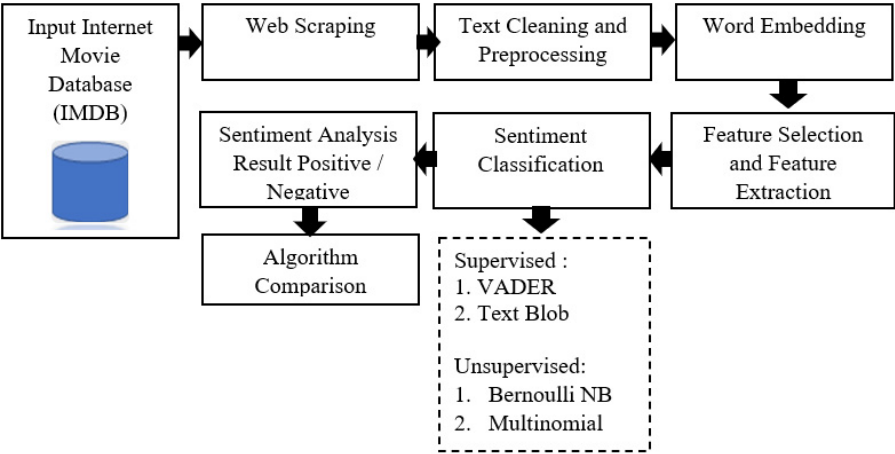


Fig. 1. Research workflow.

TFIDF model to feature selection and extractions. Hence, the TFIDF vector converts text documents to a matrix of TFIDF features.

The following methods are used to resolve the classification problem: Unsupervised Learning (VADER and Text Blob) and Supervised Learning (NB including the Bernoulli NB and Multinomial NB). Finally, the classification procedure is carried out, SA will show the positive or negative result, and the methods used are analyzed.

3.2. Internet movie database (IMDB) dataset

There are 25,000 training data, 25,000 test data, and 50,000 unlabeled data in the IMDB dataset, as seen in Fig. 2.⁴¹ The IMDB dataset is a binary sentiment classification dataset comprised of movie reviews extracted from the IMDB.⁴² Dataset

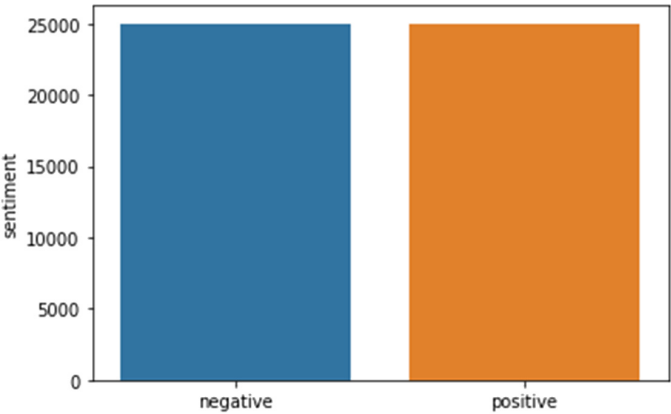


Fig. 2. Class distribution of IMDB dataset.

Table 1. Examples of movie reviews for each class.

No	Review	Sentiment
1	<i>A wonderful little production.

The filming technique is very unassuming- very old-time-BBC fashion and gives a comforting, and sometimes discomforting, sense of realism to the entire piece.

The actors are extremely well-chosen- Michael Sheen not only “has got all the polar?” but he has all the voices down pat too! You can genuinely see the seamless editing guided by the references to Williams’ diary entries; not only is it well worth the watching, but it is a terrifically written and performed piece. A masterful production about one of the great masters of comedy and his life.

The realism comes home with the little things: the fantasy of the guard, which, rather than using the traditional ‘dream’ techniques, remains solid then disappears. It plays on our knowledge and senses, particularly with the scenes concerning Orton and Halliwell, and the sets (particularly of their flat with Halliwell’s murals decorating every surface) are well done.</i>	Positive
2	<i>There’s a family where a little boy (Jake) thinks there’s a zombie in his closet and his parents are fighting all the time.

This movie is slower than a soap opera... and suddenly, Jake decides to become Rambo and kill the zombie.

OK, first of all when you’re going to make a film you must Decide if it’s a thriller or a drama! As a drama, the movie is watchable. Parents are divorcing & arguing like in real life. And then we have Jake with his closet, which ruins all the film! I expected to see a BOOGEYMAN similar movie, but instead, I watched a drama with some meaningless thriller spots.

3 out of 10 just for the well playing parents & descent dialogs. As for the shots with Jake: ignore them.</i>	Negative

training documents are highly polarized. There is a 1:1 ratio of negative to positive for labeled documents. Because the classes in the data are evenly distributed, we are able to rely on all of the performance indicators presented before. Accuracy metrics would not yield much in the way of meaningful information if, for instance, the data were extremely unbalanced (for instance, practically all positive or negative evaluations). The rationale for this is that it is mechanically easier to correctly forecast the dominant class given the numerical abundance of the class in question. The vectors for the documents are constructed using all of the documents in the collection, including train, test, and unlabeled data. Table 1 contains several samples drawn from the IMDB dataset.

4. Experiment and Result

4.1. Result of unsupervised sentiment classification

Unsupervised learning refers to a family of machine learning techniques that make use of data that does not already have truth labels attached to it. Although the IMDB dataset contains truth labels, I produce predictions with two unsupervised SA algorithms — VADER and TextBlob — to facilitate a later comparison with more well-known supervised NLP algorithms (such as NB and LSTM, for example). We calculate predictions directly for the test subset because these two methods are lexicon-based rather than learning-based. Unsupervised sentiment classification, also

known as unsupervised SA, refers to the task of automatically assigning sentiment labels (such as positive, negative, or neutral) to text data without the use of pre-labeled training data. Instead, it relies on other techniques, such as lexicons, linguistic patterns, or clustering algorithms, to determine sentiment polarity.

Valence Aware Dictionary and sEntiment Reasoner (VADER) is a popular a rule/lexicon-based sentiment analyzer.⁴³ Lexicon approaches use a pre-defined dictionary of words and phrases, which are rated for polarity and intensity. VADER’s lexicon contains about 7,500 sentiment features, which are scored each between extremely negative $[-4]$ to extremely positive $[+4]$ by a panel human expert. Words not included in the lexicon are awarded a neutral sentiment of 0. VADER also makes use of the grammatical structure of documents (e.g. punctuation, capitalization, contrasting conjunctions). To calculate the compound sentiment score of a given document, VADER calculates the sum of the sentiment of all features in that document, and normalizes to $[-1, 1]$ (source).⁴⁴

The performance result of each classifier describes in Table 2. Out-of-the-box, the results of VADER’s classification of sentiment scores with these data are very positive.^{45,46} It accurately categorizes the feeling conveyed in 70.72% of the reviews that were used for the test. The performance of VADER, on the other hand, is significantly different between evaluations that are positive and reviews that are bad. On the one hand, it accurately classifies 86.73% of all positive reviews, but on the other side, it only successfully classifies 41.95% of negative reviews. Adjusting the classifier threshold with a precision-recall curve and/or an F -score would be a good idea if one were particularly interested in either positive or negative reviews. We are going to skip this additional step because we want to keep things as simple as possible, and because the point of this exercise is to improve my accuracy performance.

TextBlob is an NLP package that provides a variety of tools.^{47,48} Some of these capabilities include parts-of-speech tagging, SA, and noun phrase extraction. Furthermore, TextBlob, just like VADER, treats individual documents as strings; hence, to compute a sentiment vector, we must loop over each document individually. `TextBlob().sentiment` gives back a namedtuple, the first element of which is the polarity (sentiment) score, which can range anywhere from 0.0 to 1.0, and the second element is the subjectivity score, which can range anywhere from 0.0 to 1.0. Because TextBlob was designed to process raw text, we base our predictions on the

Table 2. Performance result of unsupervised sentiment classification.

Items	VADER	Text Blob
Accuracy	0.7072	0.6852
Sensitivity	0.8673	0.9356
Specificity	0.5489	0.4375
Precision	0.6553	0.6219
F1	0.7465	0.7472
Roc_Auc	0.7081	0.6866

unprocessed versions of the original reviews. TextBlob has a good performance even without any tuning, like VADER. The accuracy of TextBlob is just slightly lower than that of VADER. Even more so than with VADER, we observe significant performance disparities between classes. For example, while the model accurately categorizes approximately 94% of positive reviews, it only successfully categorizes 29% of negative reviews.

4.2. Result of supervised sentiment classification

Supervised learning algorithms are machine learning algorithms that learn from labeled training data to make predictions or classify new, unseen instances. In the context of SA, supervised learning algorithms can be trained on labeled data where each text example is associated with a sentiment label (e.g. positive, negative, or neutral).

A class of straightforward supervised learning algorithms, NB relies on Bayes' theorem and makes the "Naïve" assumption that characteristics are independent of one another regardless of the value of the outcome variable. While the fact that this assumption is commonly broken in many real-world applications, NB algorithms can be effective classifiers (particularly in document categorization and spam filtering) while being poor estimators. This contradiction has a simple solution: The distribution of dependencies within a class often cancels each other out. NB classifiers are particularly fast and scalable in comparison to more sophisticated classifiers. To analyze text (words or word combinations), this NB classifier model employs a BOWs technique. However, unlike more advanced context-dependent algorithms like BERT, BOW just considers how frequently a specific word appears in a document rather than how the words themselves are arranged. Our experiment employs sklearn's *CountVectorizer*, which tallies the percentage of total tokens in the corpus that occur in each document. Punctuation marks are automatically removed by *CountVectorizer*, then we use NLTK's regex tokenizer and English *stopwords* to get rid of any remaining non-alphanumeric characters. Since the NB API in *sklearn* doesn't come with a predefined method, our experiments started by contrasting how well each algorithm performed with my model. In addition to *n*-gram, our work also tries out the vectorizer and another hyperparameter.

TF-IDF is a statistic that reflects the relevance of a word in a document. It is defined by the number of times a word appears in a document in comparison to the number of documents that contain the word. The higher the TF-IDF value, the more significant the word is in the document. Please read this post for an informative introduction. The accuracy improves by approximately 0.4% points when utilizing sklearn's *TfidfVectorizer*. This will be the last iteration of our NB model. Notably, based on Table 3, the performance of NB is significantly superior to that of TextBlob or VADER in Table 2. It correctly classifies 87.64% of reviews, which is 16.92% points greater than the performance of the VADER model. In addition, the performance of the NB model is satisfactory with both classes, properly categorizing 86% of positive reviews.

Table 3. Performance result of supervised sentiment classification.

Metrics	BernoulliNB	MultinomialNB	MultinomialNB + N -gram	MultinomialNB + TFIDF
Accuracy	0.8556	0.8608	0.872	0.8764
Sensitivity	0.8222	0.8399	0.856	0.86
Specificity	0.8886	0.8815	0.8878	0.8926
Precision	0.8795	0.8751	0.883	0.8879
F1	0.8499	0.8571	0.8693	0.8737
Roc_Auc	0.8554	0.8607	0.8719	0.8763

Figures 3 and 4 describe the positive and negative words using Word Cloud.⁴⁹ As we can see in Fig. 3, word cloud for the positive text includes fresh air, show, perform, origin, idea, drop, write, complete, etc. Our research utilized the function WordCloud (width = 1000, height = 500, max_words = 500, min_font_size = 5). The experiment



Fig. 3. Word cloud positive text.

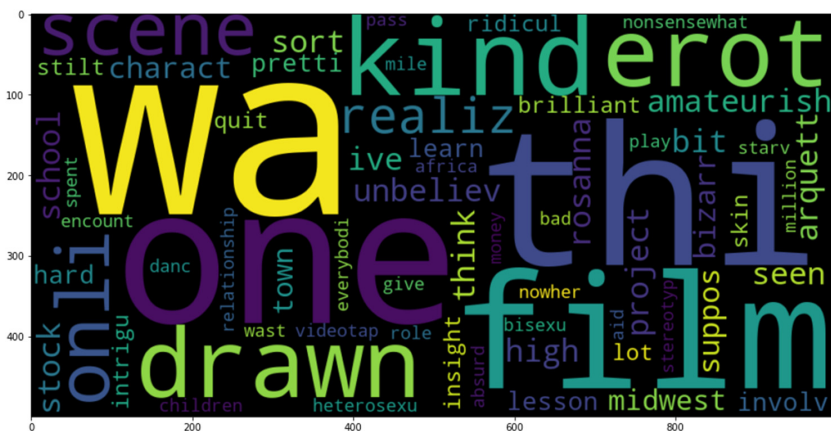


Fig. 4. Word cloud negative text.

shows the positive_words with interpolation='bilinear'. Some examples of negative text generated by word cloud are explained in Fig. 4. The negative text includes thriller, divorce, fight, parent, movie, descent, drama, kill, must, etc. Our experiment illustrates the max_words = 500 and negative_word with interpolation='bilinear'.

The following is a list of some of the benefits that the NB Algorithm offers: (1) The NB approach is effective and can cut down on the amount of time needed to complete a task significantly. (2) NB is an excellent tool for addressing problems that need the prediction of multiple classes. (3) NB has the potential to outperform other models while also using less data for training if the notion of feature independence holds true. (4) NB is better suitable for categorical input variables as opposed to numerical input variables.

5. Conclusions

The study is based on the IMDB dataset, which comprises movie reviews and the positive or negative labels that relate to them. We conducted a series of experiments to find the model with the highest accuracy and the greatest generality. The following classifiers are used in this work: Unsupervised Learning (VADER and TextBlob) and Supervised Learning (NB including the Bernoulli NB and Multinomial NB). Different techniques, including Count Vectorizer, TFIDF Vectorizer, minimum-maximum number of words, and max features, are implemented. Based on our experiment result, we can conclude the following: (1) The combination of MultinomialNB and matrix TFIDF achieves the highest accuracy, 87.64%, for both classification reports. (2) The performance of NB is significantly superior to that of TextBlob or VADER in Table 2. It correctly classifies 87.64% of reviews, which is 16.92% points greater than the performance of the VADER model. In addition, the performance of the NB model is satisfactory with both classes, properly categorizing 86% of positive reviews.

We will explore the other Sentiment Classification algorithm in future research to increase our performance results. We also want to combine SA with Shapley Additive Explanations (SHAP) for explainable artificial intelligence (XAI).

Acknowledgments

This paper is supported by the Ministry of Science and Technology, Taiwan under Grant Nos. MOST-110-2927-I-324-50, MOST-110-2221-E-324-010, and MOST-111-2221-E-324-020.

References

1. S. Kumar, M. Gahalawat, P. P. Roy, D. P. Dogra and B. G. Kim, Exploring impact of age and gender on sentiment analysis using machine learning, *Electron* 9 (IEEE, 2020) 374, doi: 10.3390/electronics9020374.

2. S. Kumar, M. Yadava and P. P. Roy, Fusion of EEG response and sentiment analysis of products review to predict customer satisfaction, *Inf. Fusion* **52** (2019) 41, doi: 10.1016/j.inffus.2018.11.001.
3. C. Dewi and R.-C. Chen, Human activity recognition based on evolution of features selection and random forest, *2019 IEEE Int. Conf. Syst. Man Cybern.* (IEEE, 6–9 October 2019), pp. 2496–2501, doi: 10.1109/SMC.2019.8913868.
4. J. H. Kim, B. G. Kim, P. P. Roy and D. M. Jeong, Efficient facial expression recognition algorithm based on hierarchical deep neural network structure, *IEEE Access* **7** (2019) 41273, doi: 10.1109/ACCESS.2019.2907327.
5. A. S. Manek, P. D. Shenoy, M. C. Mohan and K. R. Venugopal, Aspect term extraction for sentiment analysis in large movie reviews using Gini Index feature selection method and SVM classifier, *World Wide Web* **20** (2017) 135, doi: 10.1007/s11280-015-0381-x.
6. C. N. Dos Santos and M. Gatti, Deep convolutional neural networks for sentiment analysis of short texts, in *Proc. COLING 2014, 25th Int. Conf. Computational Linguistics*, 23–29 August 2014, Dublin, Ireland, pp. 69–78.
7. M. Pontiki, D. Galanis, H. Papageorgiou, S. Manandhar and I. Androutsopoulos, SemEval-2015 Task 12: Aspect based sentiment analysis, in *Proc. 9th Int. Workshop on Semantic Evaluation* (Association for Computational Linguistics, Denver, Colorado, 2015), pp. 486–495, doi: 10.18653/v1/s15-2082.
8. C. Dewi, R. C. Chen, H. Yu and X. Jiang, Robust detection method for improving small traffic sign recognition based on spatial pyramid pooling, *J. Ambient Intell. Humaniz. Comput.* **12** (2021) 1–18, doi: 10.1007/s12652-021-03584-0.
9. D. Cao, R. Ji, D. Lin and S. Li, A cross-media public sentiment analysis system for microblog, *Multimed. Syst.* **22** (2016) 479–486, doi: 10.1007/s00530-014-0407-8.
10. R. Ren, D. D. Wu and T. Liu, Forecasting stock market movement direction using sentiment analysis and support vector machine, *IEEE Syst. J.* **13**(1) (2019) 760–770, doi: 10.1109/JSYST.2018.2794462.
11. A. H. Shapiro, M. Sudhof and D. J. Wilson, Measuring news sentiment. *J. Econ.* **228**(2) (2022) 221–243, doi: 10.1016/j.jeconom.2020.07.053.
12. P. Ardianto, R. B. R. Subiakto, C. Y. Lin, Y. K. Jan, B. Y. Liao, J. Y. Tsai, V. B. H. Akbari and C. W. Lung, A deep learning method for foot progression angle detection in plantar pressure images, *Sensors* **22** (2022) 2786, doi: 10.3390/s22072786.
13. G. Abercrombie and R. Batista-Navarro, ParlVote: A corpus for sentiment analysis of political debates, *12th Int. Conf. Language Resources and Evaluation (LREC 2020)* (European Language Resources Association, 2020), pp. 5073–5078.
14. C. Dewi, R. C. Chen, Y. T. Liu, X. Jiang and K. D. Hartomo, Yolo V4 for advanced traffic sign recognition with synthetic training data generated by various GAN, *IEEE Access* **9** (2021) 97228–97242, doi: 10.1109/ACCESS.2021.3094201.
15. S. Chatterjee, K. Chakrabarti, A. Garain, F. Schwenker and R. Sarkar, Jumrv1: A sentiment analysis dataset for movie recommendation, *Appl. Sci.* **11** (2021) 9381, doi: 10.3390/app11209381.
16. C. Dewi, R.-C. Chen, Y.-T. Liu and S.-K. Tai, Synthetic data generation using DCGAN for improved traffic sign recognition, *Neural Comput. Appl.* **33** (2021) 1–15.
17. R.-C. Chen, C. Dewi, W.-W. Zhang and J.-M. Liu, Integrating gesture control board and image recognition for gesture recognition based on deep learning, *Int. J. Appl. Sci. Eng.* **17** (2020) 237–248.
18. C. Dewi, R.-C. Chen, X. Jiang and H. Yu, Deep convolutional neural network for enhancing traffic sign recognition developed on Yolo V4, *Multimed. Tools Appl.* **81** (2022) 1–25, doi: 10.1007/s11042-022-12962-5.

19. C. Dewi and R.-C. Chen, Complement Naïve Bayes classifier for sentiment analysis of internet movie database, in *Asian Conference on Intelligent Information and Database Systems*, Lecture Notes in Computer Science, Vol. 13757 (Springer, Cham, 2022), pp. 81–93, doi: 10.1007/978-3-031-21743-2_7.
20. C. Zirn, M. Niepert, M. Strube and H. Stuckenschmidt, Fine-grained sentiment analysis with structural features, in *Proc. 5th Int. Jt. Conf. Nat. Lang. Process.* (Asian Federation of Natural Language Processing, Chiang Mai, Thailand, 2011), pp. 336–344.
21. O. Appel, F. Chiclana, J. Carter and H. Fujita, Successes and challenges in developing a hybrid approach to sentiment analysis, *Appl. Intell.* **48** (2018) 1176–1188, doi: 10.1007/s10489-017-0966-4.
22. B. Pang, L. Lee and S. Vithyanathan, Thumbs up? Sentiment classification using machine learning techniques, in *Proc. 2002 Conf. Empirical Methods in Natural Language Processing (EMNLP, 2002)*. (Association for Computational Linguistics, 2002), pp. 79–86.
23. C. Dewi and R. C. Chen, Random forest and support vector machine on features selection for regression analysis, *Int. J. Innov. Comput. Inf. Control.* **15** (2019) 2027–2037, doi: 10.24507/ijicic.15.06.2027.
24. E. Redivo, C. Viroli and A. Farcomeni, Quantile-distribution functions and their use for classification, with application to Naïve Bayes classifiers, *Stat. Comput.* **33** (2023) 55, doi: 10.1007/s11222-023-10224-4.
25. S. Chen, G. I. Webb, L. Liu and X. Ma, A novel selective Naïve Bayes algorithm, *Knowl.-Based Syst.* **192** (2020) 105361, doi: 10.1016/j.knosys.2019.105361.
26. S. Kanwar, L. K. Awasthi and V. Shrivastava, Candidate project selection in cross project defect prediction using hybrid method, *Expert Syst. Appl.* **218** (2023) 119625, doi: 10.1016/j.eswa.2023.119625.
27. A. O. Salau, T. A. Assegie, A. T. Akindadelo and J. N. Eneh, Evaluation of Bernoulli Naïve Bayes model for detection of distributed denial of service attacks, *Bull. Electr. Eng. Inf.* **12** (2023) 1203–1208, doi: 10.11591/eei.v12i2.4020.
28. S. R. Shah, A. Kaushik, S. Sharma and J. Shah, Opinion-mining on marglish and devanagari comments of youtube cookery channels using parametric and non-parametric learning models, *Big Data Cogn. Comput.* **4** (2020) 3, doi: 10.3390/bdcc4010003.
29. F. Malan and J. L. Jooste, Text mining techniques for identifying failure modes, *J. Qual. Main. Eng.* **29** (2023) 666–682, doi: 10.1108/JQME-02-2020-0012.
30. C. Dewi, R. Chen, Y. Liu and H. Yu, Various generative adversarial networks model for synthetic prohibitory sign image generation, *Appl. Sci.* **11** (2021) 2913.
31. C. Dewi, R.-C. Chen and S.-K. Tai, Evaluation of robust spatial pyramid pooling based on convolutional neural network for traffic sign recognition system, *Electronics* **9** (2020) 889, doi: 10.3390/electronics9060889.
32. C. H. Lee, F. Gutierrez and D. Dou, Calculating feature weights in Naïve Bayes with Kullback-Leibler measure, in *Proc. IEEE Int. Conf. Data Mining (IEEE, 2011)*, pp. 1–12, doi: 10.1109/ICDM.2011.29.
33. N. A. Zaidi, J. Cerquides, M. J. Carman and G. I. Webb, Alleviating Naïve Bayes attribute independence assumption by attribute weighting, *J. Mach. Learn. Res.* **14** (2013) 1947–1988, doi: 10.13039/501100000923.
34. C. Dewi, R.-C. Chen, Hendry and H.-T. Hung, Experiment improvement of restricted Boltzmann machine methods for image classification, *Vietnam J. Comput. Sci.* **8** (2021) 1–16, doi: 10.1142/S2196888821500184.
35. S. Rawat, L. Werulkar and S. Jaywant, Text-based language identifier using multinomial Naïve Bayes algorithm, *Int. J. Next-Gener. Comput.* **14** (2023) 1–16, doi: 10.47164/ijngc.v14i1.1024.

36. Z. Hugh, A. Alabousi and O. Mironov, Classification of musculoskeletal radiograph requisition appropriateness using machine learning, *Can. Assoc. Radiol. J.* **74** (2023) 93–99, doi: 10.1177/08465371221121074.
37. R. C. Chen, C. Dewi, S. W. Huang and R. E. Caraka, Selecting critical features for data classification based on machine learning methods, *J. Big Data.* **7** (2020) 1–26, doi: 10.1186/s40537-020-00327-4.
38. C. Dewi, R. C. Chen and Y. T. Liu, Wasserstein generative adversarial networks for realistic traffic sign image generation, in *Intelligent Information and Database Systems*, Lecture Notes in Computer Science Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics, Vol. **12672** (Springer, Phuket, Thailand, 2021), pp. 479–493, doi: 10.1007/978-3-030-73280-6_38.
39. S. Tai, C. Dewi, R. Chen, Y. Liu, X. Jiang and H. Yu, Deep learning for traffic sign recognition based on spatial pyramid pooling with scale analysis, *Appl. Sci.* **10** (2020) 6997, doi: 10.3390/app10196997.
40. C. Dewi, R. C. Chen and H. Yu, Weight analysis for various prohibitory sign detection and recognition using deep learning, *Multimed. Tools Appl.* **79** (2020) 32897–32915, doi: 10.1007/s11042-020-09509-x.
41. N. Lakshminpathi, IMDB dataset of 50K movie reviews, <https://www.kaggle.com/datasets/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews>, last accessed 2023/01/10.
42. C. Dewi, R. C. Chen and Y.-T. Liu, Taiwan stop sign recognition with customize anchor, *12th Int. Conf. Computer Modeling and Simulation (ICCMS '20)*, 26–28 February, 2020, Brisbane, QLD, Australia, pp. 51–55.
43. S. H. Lee, H. Lee and J. H. Kim, Enhancing the prediction of user satisfaction with metaverse service through machine learning, *Comput. Mater. Contin.* **72** (2022) 4983–4997, doi: 10.32604/cmc.2022.027943.
44. R. Jain, A. Kumar, A. Nayyar, K. Dewan, R. Garg, S. Raman and S. Ganguly, Explaining sentiment analysis results on social media texts through visualization, *Multimed. Tools Appl.* **82** (2023) 22613–22629, doi: 10.1007/s11042-023-14432-y.
45. A. Vohra and R. Garg, Deep learning based sentiment analysis of public perception of working from home through tweets, *J. Intell. Inf. Syst.* **60** (2023) 255–274, doi: 10.1007/s10844-022-00736-2.
46. D. Jewel, R. E. Kelemen, R. L. Huang, Z. Zhu, B. Sundares, X. Cao, K. Malley, Z. Huang, M. Pasha, J. Anthony, T. van Opijnen and A. Chatterjee, Virus-assisted directed evolution of enhanced suppressor tRNAs in mammalian cells, *Nat. Meth.* **20** (2023) 95–103, doi: 10.1038/s41592-022-01706-w.
47. B. M. Alenzi, M. B. Khan, M. H. A. Hasanat, A. K. J. Saudagar, M. Alkhathami and A. Altameem, Automatic annotation performance of TextBlob and VADER on Covid vaccination dataset, *Intell. Autom. Soft Comput.* **34** (2022) 1311–1331, doi: 10.32604/iasc.2022.025861.
48. W. Aljedaani, F. Rustam, M. W. Mkaouer, A. Ghallab, V. Rupapara, P. B. Washington, E. Lee and I. Ashraf, Sentiment analysis on Twitter data integrating TextBlob and deep learning models: The case of US airline industry, *Knowl.-Based Syst.* **255** (2022) 109780, doi: 10.1016/j.knosys.2022.109780.
49. B. Tessem, S. Bjørnstad, W. Chen and L. Nyre, Word cloud visualisation of locative information, *J. Locat. Based Serv.* **9** (2015) 254–272, doi: 10.1080/17489725.2015.1118566.