EMITTER International Journal of Engineering Technology
Vol. 13, No. 1, June 2025, pp. 1~21
DOI : 10.24003/emitter.v13i1.821

Exploring the Power of Feature Representations: A Comparative Study on Product Reviews for Sentiment Analysis

Thian Lian Ben[1], Ravikumar R N[1], Sushil Kumar Singh[1], Sivakumar N[2], Pratikkumar Chauhan[1], Manoj Praveen V[3]

[1]Department of Computer Engineering, Marwadi University, Rajkot, India
[2]Department of Computer Science and Engineering, Alliance University, Bengaluru, India
[3]Department of AI & DS, Velalar College of Engineering and Technology, Erode, India
Corresponding Author: ravikumar.natarajan@marwadieducation.edu.in

Received January 13, 2025; Revised February 20, 2025; Accepted April 2, 2025

Abstract

With the rise of e-commerce and online shopping, customer reviews have become a crucial factor in determining the quality and reputation of a product. Online shoppers rely heavily on customer reviews to make informed purchasing decisions, as they don't have the opportunity to physically examine the product before buying. As a result, companies are also investing in sentiment analysis to understand and respond to customer feedback, as well as to enhance the quality of their products and services. Using natural language processing (NLP) and machine learning techniques, sentiment analysis classifies the tone of a customer review as positive, negative, or neutral. It involves analysing text data to determine the overall tone, emotion, and opinion expressed in a review. In this work, we study sentiment analysis of client reviews using machine learning algorithms with different vectorization techniques. The strategy outlined here consists of three distinct phases. The initial step involves some pre-processing to get rid of irrelevant information and find the useful terms. Then, feature extraction was accomplished utilizing numerous vectorization strategies as Bag-Of-Words (BoW), Term Frequency Inverse Document Frequency (TF-IDF), and N-grams. After extracting the features from text data, the final stage is classification and predictions based on machine learning approaches. We evaluated the proposed models on Yelp reviews dataset. The experimental results are evaluated using metrics such as precision, recall, and f1-score, and K-fold cross-validation.

Keywords: Sentiment Analysis, Natural Language Processing, Yelp Review Dataset, Feature Extraction, TF-IDF

1. INTRODUCTION

The widespread use of social networks as well as the internet has dramatically impacted the way products are manufactured, marketed, and sold. With a growing number of consumers shopping online, e-commerce has

become a vital channel for businesses to reach customers and sell their products. The anonymity and convenience of the internet has made it easier for people to freely express their feelings and emotions online, including on websites and social media platforms. This has resulted in a vast amount of data that can provide valuable insights into public opinion and sentiment. Websites such as Amazon, Yelp, and JD.com, to name a few, provide customers with a platform to express their opinions and share their experiences with products and services. This results in large amounts of user-generated data, including customer reviews, ratings, and comments. Businesses can utilize this information to better understand their customers' wants, needs, and preferences, which in turn improves their product design, advertising, and customer service. The fast growth of customer comments and reviews online creates a large amount of data, making it challenging for manufacturers to manually analyse them. Sentiment analysis is a useful tool for quickly and accurately detecting the sentiment polarity of customer feedback, as it combines machine learning, natural language processing, and text categorization to classify comments as positive, negative, or neutral [1]. The aim of sentiment evaluation is to better comprehend consumer feedback in order to make informed decisions on product, service, and customer experience development. Businesses can utilise sentiment analysis to learn more about client feedback and improve their offerings in response.

In recent years, the field of sentiment analysis has recently gained a lot of attention, and many researchers have turned to machine learning techniques to create sentiment analysis models. Sentiment analysis for product reviews will be considered as two classification problems in this paper, that is positive and negative emotional tendencies. And the features in text were extracted by three different techniques, which are BoW, N-grams and TF-IDF. Furthermore, we use four distinct machine learning algorithms to categorize emotional tendencies. They are Support Vector Machine (SVM), Random Forest (RF), Logistic Regression (LR) and Gradient Boosting (GB) algorithms. The models will be trained by those different algorithms on a training sample set and provide results of sentiment classification on the test set.

The objective of this work is:

- To study and apply the NLP and ML techniques to predict the sentiment based on the product reviews.
- To implement feature representation techniques.
- To explore RF, LR, GB, SVM, BoW, N-grams, and TF-IDF under different K values.
- To evaluate and compare the performances of the algorithms.

There are five parts to this paper. The first part is an introductory overview, and the second examines related literature. The third part covers the techniques used for data pre-processing and feature extraction. Moving on to section four, we will discuss the machine learning algorithms employed,

along with presenting the experimental results and analysis. Finally, the last part concludes our work.

2. RELATED WORKS

The paper generally makes use of three types of approaches: machine learning-based, hybrid-based and lexicon-based learning [2]. By comparing the ability of machine learning algorithms which were used in this paper, Random Forest Classifier performed with 80% accuracy which was better than other algorithms and Decision Tree could not perform very well with 52% of accuracy and 71% accuracy is given by K- Nearest Neighbour Classifier. Random Forest can perform 5% accuracy better than SVM using the online product reviews collected from flipkart.com [3]. SVM gives 92% accuracy and 97% of accuracy was achieved by the RF classifier. With TF-IDF for extracting the features and calculating the weight of words from tweets dataset and before performing classification techniques like SVM, Naive Bayes (NB), Ant colony and Particle swarm optimizations process was considered [4]. Comparing NB-ACO, NB-PSO, and SVM-ACO with SVM-PSO, we observe that NB-ACO achieves higher accuracy, while SVM-ACO outperforms SVM-PSO. Generally, we can see that the accuracy of SVM is better than NB. Sentiment analysis implemented using two different machine learning algorithms, namely SVM and NB on the feedback dataset of different products which is collected from Amazon website in this paper [5]. SVM and NB obtained different accuracy for each dataset and among them, 98.17% accuracy of NB is achieved for Camera reviews as well as 93.54% accuracy of SVM. To vectorize the sentences, Bag of Words is applied.

Five kinds of machine learning algorithms are used to classify the polarity of movie reviews which contain 2000 reviews where there are the same number of reviews for different labels [6]. The techniques utilized include SVM, Bernoulli Naïve Bayes (BNB), Decision Tree, Maximum Entropy (ME), and Multinomial Naïve Bayes (MNB). Each of these algorithms demonstrated varying levels of accuracy, f-score, recall and precision. Regarding accuracy, Multinomial Naïve Bayes outperforms the other techniques. A maximum of 88.50% accuracy was attained by MNB, 87.50% by Bernoulli NB, 87.33% by SVM, 60.67% by Maximum Entropy, and 80.17% by Decision Tree. Although Multinomial Naïve Bayes achieves a high f-score and precision, the recall of SVM is higher. We can see that the effectiveness of logistic regression on Twitter dataset is better than Multinomial Naive Bayes, SVM [7].

The paper proposed the comparison between the effectiveness of supervised machine learning models on Twitter data using n-gram and bi-gram models. The accuracy of the three models were evaluated, with Logistic Regression achieving a near 86.23% accuracy, SVM achieving an 85.69% accuracy of and MNB achieving an accuracy of 83.54%. An exhaustive experimental analysis of sentiment classification was conducted using a diverse range of machine learning techniques [5]. The experiments encompassed a variety of classification techniques, which included Maximum

Entropy, SVM, Random Forest, Bagging, Boosting, and Decision Tree. This experiment utilized three review datasets obtained from Amazon, Yelp, and IMDb, which pertained to diverse domains such as products, services, and entertainments. According to this experiment results, Bagging achieves the best recall with 87%, 87% of precision and f-score with 86.5% of Amazon dataset. Likewise, for Yelp review dataset Maximum Entropy can perform well with 0.760 of recall, 0.760 of precision and 0.755 with f-score. Similarly, SVM provides 0.760 of recall, 0.755 of precision and 0.755 of f-score on IMDB review dataset. For all datasets, every other machine learning classifier achieved results with accuracy close to 80%.

Machine learning algorithm called SVM is utilized for training the model for a real-time sentiment analysis on product reviews which is gathered from Amazon e-commerce website [8]. The high accuracy with approximately 87.88%, precision with 87.88%, recall as 99.98% and f-score with 93.54% are obtained in this experiment. Five methods for sentiment analysis such as lexicon based, hybrid, k-means, supervised machine learning based and k-modes using BoW approaches were explored [9]. It used five different feature extraction techniques in Tamil texts, namely as BoW, Term Frequency (TF) algorithm, TF-IDF technique, Word2Vec and fastText. Each technique obtained different results. According to the experimental results, 79% was obtained as the highest accuracy using fastText by supervised learning based approach. To determine the proper sentiment toward the actual target entity, a semantic conceptualization technique using tagged bags of concepts (TBoC) for sentiment analysis is provided [10].

This approach considers the emotional and intellectual content of the writing, with a focus on the concise text. NB, Neural Network(NN) and SVM are applied for sentiment classification. Two strategies have been used to implement the TBoC approach. SentiWordNet was used for polarity detection in the first method named TBoC (SWN), while a domain-specific sentiment lexicon was used in the second method named TBoC (DSL). The best accuracy results were generated by NB and NN, but TBoC (DSL) was not far behind. SVM technique performance was the poorest of all. Only NB and NN have recall rates that are greater than 75%, while NB has the highest precision rate of 77%. Out of all the methods evaluated, only TBoC (DSL) and NB were able to achieve accuracy results above 70%. TBoC (DSL) outperformed all other methods across all evaluation metrics, achieving over 75% average precision and 73% average recall. Meanwhile, the results obtained using TBoC (SWN) were comparable to those obtained using other state-of-the-art methods.The study utilized hybrid deep learning models that incorporated TF-IDF weighted Glove word embeddings [11]. We can see that the empirical analysis includes the effectiveness of prediction using various word embedding techniques, including word2vec and other techniques, with weighted word embedding techniques such as TF-IDF, IDF and Smoothed inverse document frequency (SIF). When we compare the experimental

results, the weighted word embedding system gives better performance than unweighted word embedding system.

3. ORIGINALITY

We performed sentiment classification using various feature representation techniques through supervised learning algorithms. Four different kinds of machine learning algorithms of supervised learning as follows RF classifier, LR, GB and SVM are utilized. Moreover, BoW, N-grams and TF-IDF vectorization techniques are employed in each of the proposed machine learning algorithms.

It is possible that the results of an evaluation of a machine learning model trained on a single dataset are not representative of its overall performance, and therefore it is often necessary to train the model on multiple datasets. To avoid this problem, we divided our source data into two divisions:

- Train (80%): Used in the learning process to feed the machine learning algorithm.
- Test (20%): Used to improve generalization and minimize overfitting problem.

4. SYSTEM DESIGN

The following section shows the system design such as Data Collection, Data Pre-processing, Feature Extraction methods such as Bag-of-words, TF-IDF, N-grams, Classification Techniques are applied on Yelp review dataset.

4.1 Data Collection

Data collection involves recording past events so that data analysis can be performed to identify recurring trends. By using those records, we built the models using machine learning algorithms that predict how things will change in the future. The collection of high-quality data is therefore crucial.

The proposed system collects product reviews from the Yelp Dataset Challenge 2015 data. The product reviews polarity dataset is built by taking into account stars 1 and 2 as negative, and 3 and 4 as positive. Even though there are many samples in the dataset, the proposed system only selects 32,000 training and 8,000 testing are drawn at random. Therefore, there are a total of 40,000 samples. Class 1 describes positive polarity, whereas Class 2 describes negative polarity.

4.2 Data Pre-processing

Before extracting the features from the input text, data pre-processing is employed first because the reviews may include meaningless words, emoji, icons and improper words. In the field of machine learning, data preparation involves transforming (normalizing and cleaning) the review data into something useful that can be applied to build and train machine learning models [12]. In this stage, there are some common preparation/cleaning steps:

1) Removal of html tags
2) Converting emoticons to words
3) Removal of punctuations
4) Tokenization
5) Part-of-Speech tagging
6) Removal of stop-words
7) Converting words to lowercase
8) Lemmatization

The proposed system removes HTML tags and emoticons as they don't serve the meaning of review. Tokenization, which converts emoticons to words, helps the machine understand human emotions. The system then removes punctuation and tokenization, which divides a written document into smaller components, which is crucial for sentiment analysis. Tokenization is an essential step in pre-processing text data [13].

Then, POS tagging stands for Part-of-Speech tagging. One must label each word according to its grammatical role, with categories like noun, verb, adjective, adverb, and others used to describe its function within a sentence. Removing stop words involves cleansing the text data by removing common words that do not convey any useful information. These words are known as stop words and they include words such as "the", "a", "an", "and", "or", "but", etc. Stop words are typically removed from text data before performing sentiment analysis because they do not contribute much to the overall meaning of the text and can sometimes even cause noise in the data. Removing stop words can help in order to flatten the textual data and make it easier to analyze and process [14].

Sentiment analysis removes irrelevant words from text data to improve accuracy and focus on meaningful words and phrases. Converting all text data to lowercase minimizes duplicate words and improves analysis capabilities. This reduces the size of the vocabulary used in the analysis and improves the accuracy of results. Lemmatization reduces words to their bases to normalize text data and discern emotional tone. It simplifies text emotion detection by reducing word forms to a single base form. [15].

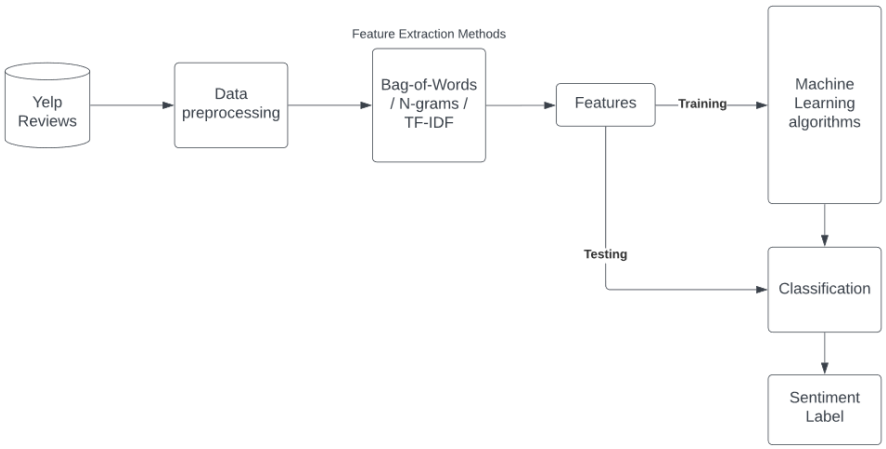

Figure 1. Flow of the study works.

By reducing words to their base form, the sentiment analysis algorithm can treat words with the same meaning, regardless of their inflectional forms, as the same word, as shown at Figure 1. This can help to minimize the size of the vocabulary used in the analysis and develop accuracy.

4.3 Feature Representation Techniques

We have applied three feature representation techniques such as Bag of Words, TF-IDF, and N-grams.

4.3.1 Bag-of-Words

Bag-of-Words (BoW) is a common technique utilized in NLP tasks, including sentiment analysis. The basic idea behind BoW is to represent a piece of text as a vector of word counts. The process of creating a BoW vector typically involves the following steps:

1. Tokenization: The initial step is for breaking the text into individual words or tokens. This is often done by splitting the text on spaces and punctuation marks.

2. Vocabulary creation: Next, vocabulary is created by collecting all the unique words in the text. This vocabulary serves as the set of features for the BoW representation.

3. Counting: The text is converted into a numerical vector that records how often some words appear in a vocabulary, with each vector element representing a unique word [16].

4. Normalization: The word count vectors can be normalized, to make it less sensitive to the text length.

5. Encoding: Once the vectors have been encoded, they can be fed into the classification algorithm as input.

BoW is simple, efficient and widely used in NLP tasks, especially in sentiment analysis [17]. It records the occurrences of terms throughout the text, which can be quite helpful to determine the overall mood [18]. However, it does not capture the context of the words, and it can be sensitive to the text length.

For example, let's say we have two sentences:

"I love this product" and "I hate this product"

If we use a BoW representation, the vector for each sentence will be almost identical, but the sentiment conveyed in each sentence is quite different.

4.3.2 TF-IDF

TF-IDF, is a measure of a word's relevance to a corpus or a set of texts [19]. The frequency of a term in the corpus is used to counteract the increase in importance that occurs when a word appears frequently in the dataset.

Term Frequency, named as TF, represents how often certain words appear in a given document d [20]. In document d, the frequency shows how frequently a word, *t*, is used. Simply put, how often a word or phrase appears

in a given text determines its weight. *tf(t,d)* represents how many times *t* shows up in document d [21].

$$tf(t,d) = \frac{raw\ count\ of\ a\ term\ t\ in\ a\ document}{total\ number\ of\ terms\ t\ in\ document} \quad (1)$$

Document Frequency (DF) is how many times the word *t* shows up in the whole document d, denoted by *df(t).* A measure of how much information a word conveys, or whether it is frequent or uncommon across all documents, known as the inverse document frequency or IDF [20]. A document's IDF can be determined by dividing its frequency by the total number of documents in the corpus.

$$idf(t,d) = log(\frac{number\ of\ documents\ in\ collections\ (N)}{df(t)}) \quad (2)$$

The following formula is used to calculate TF-IDF.

$$Tf.IDF = tf(t,d) \times idf(t,d) \quad (3)$$

To implement TF-IDF, train and test data goes to the data pre-processing step.

4.3.3 N-grams

N-grams may also help identify document word and phrase cluster frequencies. It breaks a text into n-word substrings [22]. For example: “The movie was fantastic. The acting was superb”. The N-grams representation for one example sentence is listed in Table 1.

Table 1. N-grams representation for one example sentence

Trigram	Count
The movie was	1
movie was fantastic	1
was fantastic The	1
fantastic The acting	1
The acting was	1
acting was superb	1

By keeping track of how often each chunk appears in a given text, we may build a list of numbers that accurately represents the text. The ability of N-grams to grasp the structure and context of a text makes them useful. The

number of possible n-grams, however, might get rather huge when n is large, or the text corpus is enormous [23]. To address this challenge, model performance can be improved by employing pruning and smoothing techniques to reduce the number of features. Trigram was used to extract textual characteristics in this investigation.

4.4 Classification Techniques

We experimented with different classifiers such as Random Forest, Support Vector Machine, Logistic Regression, and Gradient Boost.

4.4.1 Random Forest Classifier

The Random Forest (RF) Classifier is a machine learning algorithm used for classification tasks. It extends the RF algorithm, which is an ensemble learning technique that uses a network of interconnected many decision trees to produce a more reliable and precise prediction [24]. In the RF algorithm, the input data is split into multiple subsets, each of which is used to train a decision tree. Each decision tree in the RF Classifier algorithm is trained using a unique set of input features and a random subset for the training data to increase its diversity and reduce overfitting [25]. Adopting this strategy aids in the reduction of overfitting and making the model more robust to noise in the input data. Once the decision trees are trained, the last prediction is made by combining the predictions of all each decision trees. This can be done using a majority vote or by averaging the predicted probabilities. The RF algorithm can also compute feature importance scores, which indicate how much each input feature contributes to the final prediction.

Here is the formula for the RF Classifier algorithm: Initialize the number of decision trees (n_trees) and the size of the random feature subset (m).

For each decision tree:

a. Sample a random subset of the training data.
b. Sample a random subset of the input features of size m.
c. Train a decision tree on the sampled data and features.

For each new input data point:

a. Evaluate the input data point using each decision tree in the ensemble.
b. Compute the last prediction by merging the predictions of all the decision trees (e.g., using a majority vote or averaging the predicted probabilities).

4.4.2 Support Vector Machine

SVM is an approach to classification algorithms which can be used to determine positive or negative sentiments. It is also used for regression problems [8]. After the process of transforming texts into vectors using text vectorization models, the algorithm helps to find the best decision boundary

between the vectors that associate with a given group and vectors that do not associate with it [26]. The SVM's optimal decision boundary is known as the hyperplane, whose dimensions are defined by the dataset's features. In the case of two features, the hyperplane will be a straight line, while for three features, it will be a two-dimensional plane [27].

SVM outperforms several other machine learning algorithms for sentiment analysis [28]. The majority of the text can be linearly separated; therefore, sentiment analysis performs quite well [29]. However, SVM is running very slow due to the numbers of support vectors if there are an excessive number of training samples.

4.4.3 Logistic Regression

Logistic Regression is an algorithm for supervised learning that can be operated in sentiment analysis to classify text into different sentiment categories (e.g. positive, negative, neutral). The fundamental concept of Logistic Regression involves utilizing a linear equation to establish the association between the input features, such as the text in the context of sentiment analysis, and the resulting output label, which represents the sentiment [30].

The linear equation is then passed through a sigmoid function (also known as a logistic function) to produce a probability value between 0 and 1, which can be comprehended as the likelihood that the input belongs to a particular class. The labelled dataset is used to train logistic regression models, where the input features are the text, and the output labels are the sentiment. The model comes to understand how the input characteristics and the output labels interact.

4.4.4 Gradient Boost Classifier

To create a more robust learner, the Gradient Boosting (GB) Classifier combines the strengths of multiple less effective learners. In this scenario, categorization is a key component of many machine learning applications. Decision trees are incorporated into the model gradually as part of the method's operation. We train each new generation of trees to correct their predecessors' errors. An objective function is optimized at each stage of the training process. Once it is successful, it is necessary to minimize the objective function's measurement of differences between predicted and actual values. The algorithm changes the predicted values so that they converge to a decreasing gradient of the objective function at each iteration [31, 32]. The accuracy of the model is increased by the algorithm as this process is repeated numerous times.

In the field of sentiment classification, GB can accurately predict how an individual would feel about a piece of words, like a movie review or social media post. The algorithm returns a probability score for each sentiment category, based on how likely it is that the text belongs to that category.

4.4.5 Evaluation Metrics

Users often express their opinions and feelings about a particular product in text. Most customers give their reviews with short sentences. Those sentences contain important words that evoke feelings about the products. In English, there are many structures of sentences and a word's meaning can vary depending on how the sentence is constructed. For example, we may observe that a term has a good meaning. but when we add un/dis/not to it, its meaning changes to a negative one. Emojis are still used by certain people to convey their emotions.

Therefore, identifying the sentiments on product reviews whether positive or negative is a challenging task. In this work, we investigate the impact of various vectorization methods on identifying the polarity of reviews. F-score, precision, and recall, which are four evaluation metrics, are used to measure the success of the classification process in this experiment. And k-fold cross-validation was utilized to evaluate the performance of the models with different feature representation techniques.

4.4.6 Confusion Matrix

The confusion matrix aids us in recognizing a model's correct predictions as well as mistakes for certain specialized classes. The matrix has four primary components that each display a distinct metric for counting the number of accurate and inaccurate predictions. We can see that each component consists of two words such as True/False and Positive/Negative. The predicted labels in the matrix are represented as positive and negative.

- The percentage of times a model correctly predicted a positive class is known as its True Positive (TP) rate.
- The number of times a model mistakenly predicts a positive class is known as the false positive rate (FP).
- The term "True Negative" (TN) describes the percentage of times a model accurately predicted a negative class.
- The rate at which a model mistakenly predicts a negative class is known as the False Negative (FN) rate.

Table 2. Confusion matrix (CF) for classification models

Actual/ Predicted	**Positive**	**Negative**
Positive	TP	FP
Negative	FN	TN

Table 2 shows the Confusion Matrix for classification models. Precision is the ability of a machine learning model to reliably identify positive instances in a given dataset. It is defined as the number of TP divided by the sum of TP and FP. For example, in a sentiment analysis task, precision

would represent the proportion of texts that were predicted as positive by the model and were actually positive.

Precision can be calculated using the following formula:

$$Precision = \frac{TP}{TP + FP} \quad (4)$$

Precision is a commonly used performance metric in machine learning, especially in problems where the cost of False Positives is high, such as in spam filtering or medical diagnosis. A high precision indicates that the model is good at avoiding False Positives, while a low precision suggests that the model is making many incorrect positive predictions.
Recall is also known as True Positive Rate (TPR) or Sensitivity. It is the number of true positive cases that are correctly labelled as positive. It is worked out as:

$$Recall = \frac{TP}{TP + FN} \quad (5)$$

The F-score evaluates a test's accuracy by considering both its precision and recall. It represents an average of recall and precision, and it is described as:

$$F - score = \frac{2 * Precision * Recall}{Precision + Recall} \quad (6)$$

4.4.7 K-fold Validation

In this study, K-fold cross-validation is utilized to evaluate the model's efficacy. It's a method of statistics for testing how well a ML model performs on a small dataset. The fundamental concept is to divide the dataset into K equal parts or "folds", where K is a positive integer. One of the K folds is set aside as the unseen set, and the other K-1 folds are added together to make the training set. The model is trained on the K-1 training folds, and the test fold is used to see how well it works. This process is repeated K times, and each of the K folds is used once as a part of the unseen set. The accuracy of the model is assessed on K different splits of the data, with each split serving as a validation set once, to obtain a more dependable estimate of its generalization ability. We applied 4-folds cross-validation in this experiment.

5. EXPERIMENT AND ANALYSIS

This subsection describes and analyzes the experimental results for our approaches. Four different machine learning algorithms and three kinds of vectorization techniques were tested on product reviews dataset obtained from the Yelp Dataset Challenge 2015 data. The dataset contains 40,000 instances of review, including 20,000 positive samples and 20,000 negative instances.

We implemented the classification of sentiment reviews on products using different machine learning techniques and comparing their results

using a confusion matrix comprising metrics like f-score, precision, recall, and average accuracy across folds.

5.1 Analysis of K-fold Validation

Our study tested machine learning techniques using Yelp review data. We used 4-folds cross-validation, calculated segment accuracy, and averaged the findings. The results of the experiment are presented in Tables 3, 4 and 5, which display the accuracy outcomes of different algorithms with different vectorization techniques. Comparing our findings to other datasets and vectorization methods confirmed their validity. The top performance in each experiment fold is bold.

Table 3. Accuracy results for each fold using TF-IDF

Algorithm	RF	LR	SVM	GB
k = 1	0.785978597	0.850485048	**0.89658965896**	0.8413841384138
k = 2	0.794079407	0.853485348	**0.90309030903**	0.8466846684668
k = 3	0.785378537	0.852485248	**0.89778977897**	0.8408840884088
k = 4	0.770877087	0.850285028	**0.89888988898**	0.8422842284228
Mean	0.784078407	0.851685168	**0.89908990899**	0.8428092809280

Table 4. Accuracy results for each fold using BoW

Algorithm	RF	LR	SVM	GB
k = 1	0.80788078807	0.8989898989	**0.90569056905**	0.84278427842
k = 2	0.78027802780	0.9009900990	**0.90519051905**	0.84738473847
k = 3	0.77747774777	0.8941894189	**0.90149014901**	0.84228422842
k = 4	0.78447844784	0.8946894689	**0.89648964896**	0.83808380838
Mean	0.78752875287	0.89721472147	**0.90221522152**	0.84263426342

Table 5. Accuracy results for each fold using N-grams.

Algorithm	RF	LR	SVM	GB
k = 1	0.8084808480	0.898289828	**0.91049104910**	0.84738473847
k = 2	0.7833783378	0.902590259	**0.90919091909**	0.8439843984
k = 3	0.8135813583	0.9032903290	**0.91509150915**	0.84388438843
k = 4	0.7676767676	0.9042904290	**0.91189118911**	0.84728472847
Mean	0.7932793279	0.9021152115	**0.91166616661**	0.84563456345

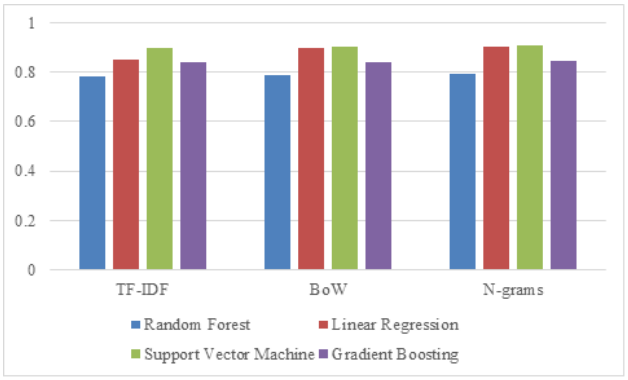

Figure 2. Comparison of Average values for each fold based on FE

Based on Figure 2, we can see that the SVM technique consistently outperforms the other techniques (RF, LR, and GB on the dataset for all three feature representation methods (TF-IDF, BoW, and N-grams). Using SVM with N-grams as the feature extraction method produces the highest accuracy score of 0.91, followed closely by SVM with Bag of Words (0.90) and LR with N-grams (0.90).

5.2 Analysis of Confusion Matrix

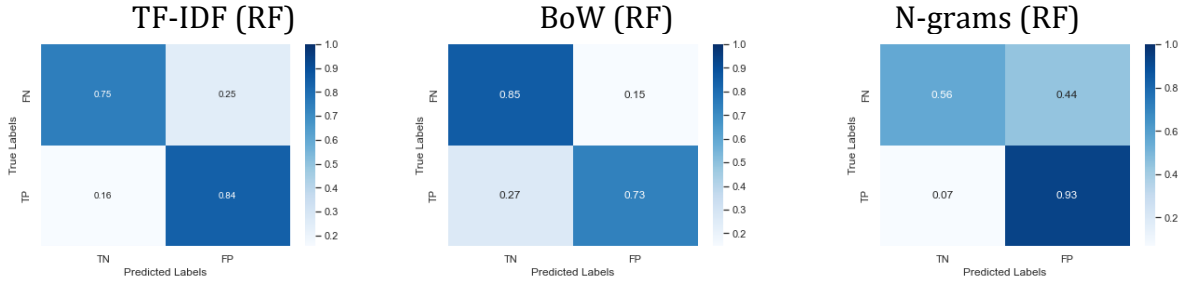

Figure 3. Confusion matrix of RF algorithm with three text representations

In Figure 3, the BoW model performs the best overall with the greatest true positive and the least false negatives. The TF-IDF model also performs well with a high TP but has a relatively high FP. The N-gram model performs the worst overall, with a low TP and a high FN.

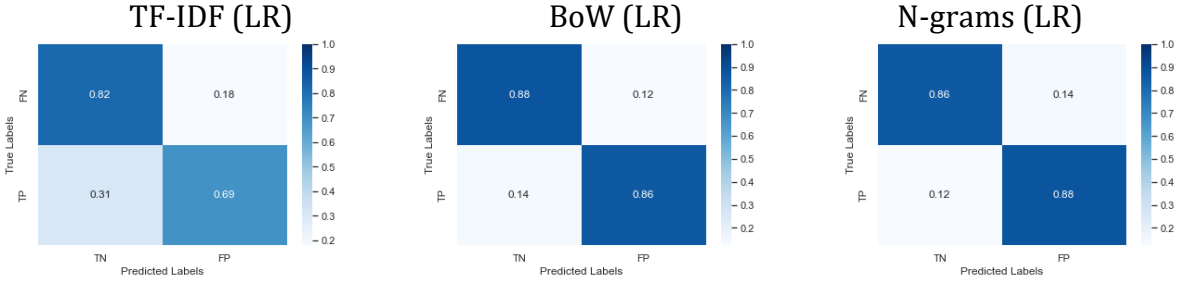

Figure 4. Confusion matrix of LR algorithm with three text representations

All three types of feature extraction techniques performed similarly well with LR algorithm in Figure 4. However, in terms of performance, the BoW feature extraction technique seems to have the highest accuracy.

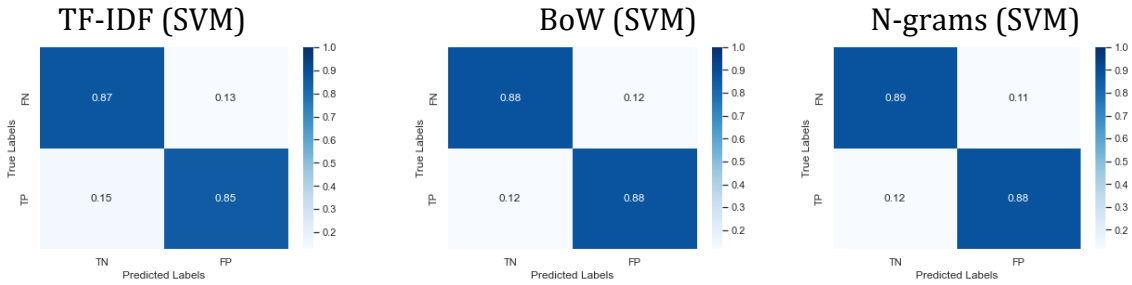

Figure 5. Confusion matrix of SVM algorithm with three text representations

Based on the results provided in Figure 5, all three feature representation techniques perform similarly well with SVM. Additionally, the performance of SVM seems to be consistent across all three types of feature representation, as the confusion matrices have similar levels of accuracy.

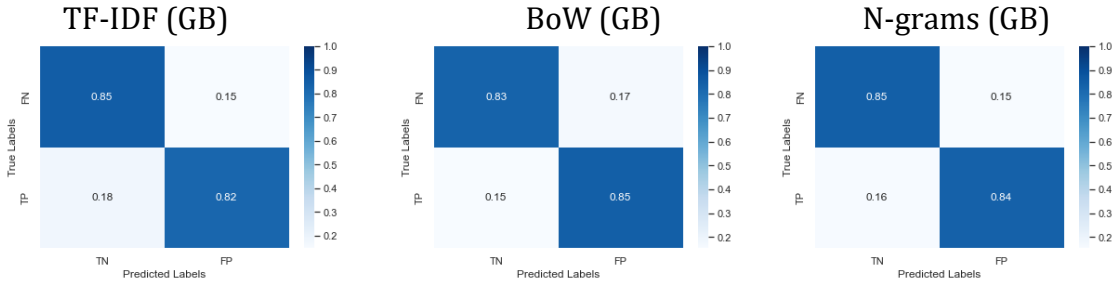

Figure 6. Confusion matrix of GB algorithm with three text representations

Meanwhile, based on the results of the GB algorithm applied to three different feature representations in Figure 6. N-grams had the highest TP rate and lowest FP rate among the three feature representations, suggesting it may be best for sentiment analysis on the dataset.

5.3 Analysis of F-score, Precision and Recall

In this section, the analysis of F-score, Precision and Recall for TF-IDF, BoW, N-grams techniques with RF, SVM, LR and GB algorithms is shown at Table 6, 7, and 8.

Table 6. F-score, recall and precision with TF-IDF technique.

	TF-IDF		
Algorithm	**F-score**	**Recall**	**Precision**
RF	0.80	0.795	0.80
SVM	0.86	0.86	0.865
LR	0.755	0.755	0.76
GB	0.835	0.83	0.835

Table 7. F-score, recall and precision for BoW technique

	Bag-of-Words		
Algorithm	**F-score**	**Recall**	**Precision**
RF	0.795	0.79	0.80
SVM	0.885	0.88	0.88
LR	0.875	0.87	0.87
GB	0.84	0.84	0.84

Table 8. F-score, recall and precision with N-grams technique.

	N-grams		
Algorithm	**F-score**	**Recall**	**Precision**
RF	0.74	0.745	0.785
SVM	0.885	0.885	0.885
LR	0.87	0.87	0.87
GB	0.845	0.84	0.845

When comparing the performance of the three methods (TF-IDF, BoW, and N-grams) on Yelp dataset, we can see that BoW and N-gram generally produce higher scores than TF-IDF as shown at Figure 7. This suggests that these techniques may be more effective for the Yelp dataset and problem being analyzed.

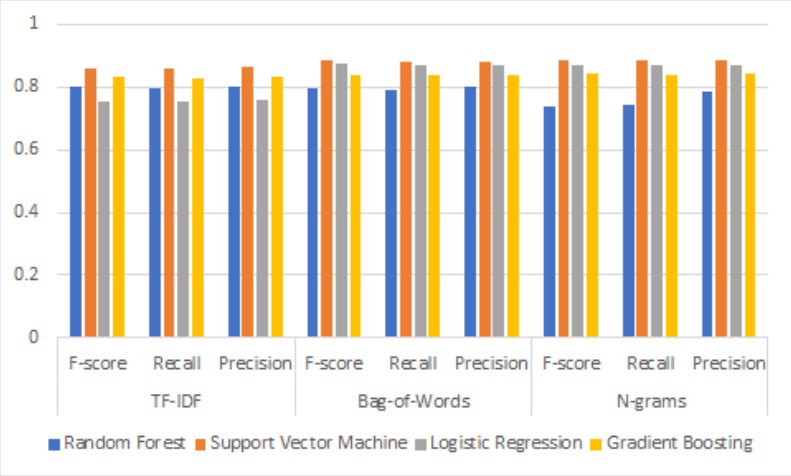

Figure 7. F-score, recall and precision with different FE techniques.

The performance of the four algorithms (RF, LR, SVM, and GB) was compared using the TF-IDF, BoW, and N-grams feature vectorization methods. Based on the average accuracy data for 4-folds, SVM was the most accurate of the three techniques for representing features. LR also did well, with both BoW and N-grams giving good accuracy. RF and GB also got reasonably good results in terms of accuracy.

Based on confusion matrix results, SVM and GB classifiers generally perform better than RF and LR classifiers, regardless of the feature representation used (TF-IDF, BoW, N-grams). However, there are some variations in performance between the different feature representations, and the differences in performance between the classifiers are not consistently large. In general, BoW and N-grams tend to perform more effectively than TF-IDF across most algorithms. The differences in performance between the classifiers are generally not large, with most confusion matrices showing relatively balanced true positives and true negatives, and relatively low false positives and false negatives.

Based on f-measure results, we can see that SVM achieved the highest F1-score with all three feature representation techniques, followed by LR, and GB. RF had a lower F1-score than the other algorithms. The recall results show similar trends to the F1-score results, with SVM achieving the highest recall scores. As reported by precision results, SVM had the highest precision with all three feature representation techniques, followed by LR and GB. Random Forest had lower precision scores than the other algorithms. As the results, SVM and LR performed consistently well across all three feature representation techniques, while GB also performed well but with slightly lower scores. RF had lower scores compared to the other algorithms.

The results suggest that all three feature extraction techniques are effective for sentiment classification tasks, with N-grams and BoW achieving slightly higher accuracy scores than TF-IDF.

6. CONCLUSION

With the growth of internet usage, the importance of understanding customer feedback and sentiment has increased. Sentiment analysis is a NLP technique that leverages artificial intelligence technology to identify and take out the subjective information conveyed in textual data, such as attitudes, emotions and opinions. This paper reports the experimental findings of sentiment analysis, in which various machine learning algorithms and vectorization techniques were employed to analyze and classify textual data based on the sentiment conveyed. Different vectorization methods such as TF-IDF, BoW and N-grams are utilized in this study. Four different machine learning algorithms, including SVM, GB, LR and RF are applied in this experiment. In this experiment, the dataset is collected from Yelp Dataset Challenge 2015 data. The models evaluated and utilized to classify sentiment polarity by utilizing the Yelp review dataset.

To evaluate the models' performance, accuracy and performance factors such as K-fold cross-validation and evaluation metrics were taken into account for each distinct vectorization technique. According to the study's results, SVM and LR algorithms showed the best performance overall for sentiment classification tasks using the three feature extraction techniques. However, the GB and RF algorithms also showed promising results and may be worth considering depending on the specific task and dataset. Additionally, all three feature extraction methods are useful for sentiment classification tasks, however N-grams and Bag-of-Words perform slightly more effectively than TF-IDF. The outcomes were then analyzed and compared to one another to determine the optimal vectorization technique for the sentiment analysis task.

Emojis and emoticons can carry emotional context and sentiment in a text message and can play a crucial role in sentiment analysis, as they can convey the emotional tone of a message. For sentiment analysis, emojis and emoticons can be treated as separate categories or be mapped to sentiment labels (e.g; positive, negative, neutral). For further work, we will extend the work with emojis and emoticons to give more accuracy than the present work.

REFERENCES

[1] S. Kausar, X. Huahu, M. Y. Shabir, and W. Ahmad, **"A Sentiment Polarity Categorization Technique for Online Product Reviews,"** *IEEE Access*, vol. 8, pp. 3594–3605, 2020.

[2] A. Mitra, **"Sentiment Analysis Using Machine Learning Approaches (Lexicon based on movie review dataset),"** *J. Ubiquitous Comput. Commun. Technol.*, vol. 2, no. 3, pp. 145–152, 2020.

[3] E. M., M. Abdul, M. Ali, and H. Ahmed, **"Social Media Sentiment Analysis using Machine Learning and Optimization Techniques,"** *Int. J. Comput. Appl.*, vol. 178, no. 41, pp. 31–36, 2019.

[4] M. Kabir, M. M. J. Kabir, S. Xu, and B. Badhon, **"An empirical research on sentiment analysis using machine learning approaches,"** *Int. J. Comput. Appl.*, vol. 43, no. 10, pp. 1011–1019, 2021.

[5] Y. S. Mehanna and M. Bin Mahmuddin, **"A Semantic Conceptualization Using Tagged Bag-of-Concepts for Sentiment Analysis,"** *IEEE Access*, vol. 9, pp. 118736–118756, 2021.

[6] A. Onan, **"Sentiment analysis on product reviews based on weighted word embeddings and deep neural networks,"** *Concurr. Comput. Pract. Exp.*, vol. 33, no. 23, pp. 1–12, 2021.

[7] T. Shaik, X. Tao, C. Dann, H. Xie, Y. Li, and L. Galligan, **"Sentiment analysis and opinion mining on educational data: A survey,"** *Nat. Lang. Process. J.*, vol. 2, no. Yan Li, p. 100003, 2023.

[8] M. Makrehchi and M. S. Kamel, **"Automatic extraction of domain-specific stopwords from labeled documents,"** *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 4956 LNCS, pp. 222–233, 2008.

[9] R. Ahuja, A. Chug, S. Kohli, S. Gupta, and P. Ahuja, **"The Impact of Features Extraction on the Sentiment Analysis,"** *Procedia Comput. Sci.*, vol. 152, pp. 341–348, 2019.

[10] B. Bansal and S. Srivastava, **"Lexicon-based Twitter sentiment analysis for vote share prediction using emoji and N-gram features,"** *Int. J. Web Based Communities*, vol. 15, no. 1, pp. 85–99, 2019.

[11] M. A. Fauzi, **"Random forest approach fo sentiment analysis in Indonesian language,"** *Indones. J. Electr. Eng. Comput. Sci.*, vol. 12, no. 1, pp. 46–50, 2018.

[12] A. S. Neogi, K. A. Garg, R. K. Mishra, and Y. K. Dwivedi, **"Sentiment analysis and classification of Indian farmers' protest using twitter data,"** *Int. J. Inf. Manag. Data Insights*, vol. 1, no. 2, p. 100019, 2021.

[13] A. Alsaeedi and M. Z. Khan, **"A study on sentiment analysis techniques of Twitter data,"** *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 2, pp. 361–374, 2019.

[14] R. Xia, C. Zong, and S. Li, **"Ensemble of feature sets and classification algorithms for sentiment classification,"** *Inf. Sci. (Ny).*, vol. 181, no. 6, pp. 1138–1152, 2011.

[15] T. Joachims, "**Text categorization with Support Vector Machines: Learning with many relevant features BT - Machine Learning: ECML-98,"** 1998, pp. 137–142.

[16] 1Ravikumar . R N., S. . Jain, and M. . Sarkar, **"Efficient Hybrid Movie Recommendation System Framework Based on A Sequential Model"**, Int J Intell Syst Appl Eng, vol. 11, no. 9s, pp. 145–155, Jul. 2023.

[17] A. Sharma and U. Ghose, **"Toward Machine Learning Based Binary Sentiment Classification of Movie Reviews for Resource Restraint Language (RRL)—Hindi,"** in IEEE Access, vol. 11, pp. 58546-58564, 2023.

[18] Kalasalingam Academy of Research and Education. IEEE Student Branch., Institute of Electrical and Electronics Engineers, and IEEE Power & Energy Society, *IEEE International Conference on Intelligent Techniques in Control, Optimization & Signal Processing : INCOS-'19 : 11th-13th April 2019.* .

[19] Rahman and M. S. Hossen, **"Sentiment Analysis on Movie Review Data Using Machine Learning Approach,"** *2019 Int. Conf. Bangla Speech Lang. Process. ICBSLP 2019*, pp. 27–28, 2019.

[20] Sri Eshwar College of Engineering and Institute of Electrical and Electronics Engineers, *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS).* .

[21] J. Jabbar, I. Urooj, W. Junsheng, and N. Azeem, **"Real-time sentiment analysis on E-Commerce application,"** *Proc. 2019 IEEE 16th Int. Conf. Networking, Sens. Control. ICNSC 2019*, pp. 391–396, 2019.

[22] S. Thavareesan and S. Mahesan, "**Sentiment Analysis in Tamil Texts: A Study on Machine Learning Techniques and Feature Representation,"** *2019 IEEE 14th Int. Conf. Ind. Inf. Syst. Eng. Innov. Ind. 4.0, ICIIS 2019 - Proc.*, pp. 320–325, 2019.

[23] G. Gautam and D. Yadav, **"Sentiment analysis of twitter data using machine learning approaches and semantic analysis,"** in *2014 Seventh International Conference on Contemporary Computing (IC3)*, 2014, pp. 437–442.

[24] J. Plisson, N. Lavrac, and D. D. Mladenić, **"A rule based approach to word lemmatization,"** *Proc. 7th Int. Multiconference Inf. Soc.*, pp. 83–86, 2004, [Online]. Available: http://eprints.pascal-network.org/archive/00000715/.

[25] R. Srivastava, P. K. Bharti, and P. Verma, **"Sentiment Analysis using Feature Generation And Machine Learning Approach,"** in *2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*, 2021, pp. 86–91.

[26] A. A. Farisi, Y. Sibaroni, and S. Al Faraby, **"Sentiment analysis on hotel reviews using Multinomial Naïve Bayes classifier,"** *J. Phys. Conf. Ser.*, vol. 1192, no. 1, 2019.

[27] G. M. Raza, Z. S. Butt, S. Latif, and A. Wahid, **"Sentiment Analysis on COVID Tweets: An Experimental Analysis on the Impact of Count Vectorizer and TF-IDF on Sentiment Predictions using Deep Learning Models,"** *2021 Int. Conf. Digit. Futur. Transform. Technol. ICoDT2 2021*, 2021.

[28] V. Sundaram, S. Ahmed, S. A. Muqtadeer, and R. R. Reddy, **"Emotion Analysis in Text using TF-IDF,"** in *2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, 2021, pp. 292–297.

[29] S. M. H. Dadgar, M. S. Araghi, and M. M. Farahani, **"A novel text mining approach based on TF-IDF and Support Vector Machine for news classification,"** in *2016 IEEE International Conference on Engineering*

and Technology (ICETECH), 2016, pp. 112–116.

[30] S. Kaur, G. Sikka, and L. K. Awasthi, **"Sentiment Analysis Approach Based on N-gram and KNN Classifier,"** in *2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)*, 2018, pp. 1–4.

[31] M. Aufar, R. Andreswari, and D. Pramesti, **"Sentiment Analysis on Youtube Social Media Using Decision Tree and Random Forest Algorithm: A Case Study,"** *2020 Int. Conf. Data Sci. Its Appl. ICoDSA 2020*, 2020

[32] K. Zahoor, N. Z. Bawany, and S. Hamid, **"Sentiment analysis and classification of restaurant reviews using machine learning,"** *Proc. - 2020 21st Int. Arab Conf. Inf. Technol. ACIT 2020*, 2020.