AN ENSEMBLE METHOD FOR SENTIMENT ANALYSIS ON TEXTILE DATASET USING MACHINE LEARNING ALGORITHMS

S. Sheela Devi[1], Dr. Viji Vinod[2]

[1]Research Scholar, E-Mail: sheelababubca12@gmail.com[1]
[2]Professor & Head, Department of Computer Applications,Dr. M.G.R Educational and Research Institute, Maduravoyal, Chennai, Tamilnadu, India , vijivinod.mca@drmgrdu.ac.in[2.]

KEYWORDS

Naive Bayes, Support Vector Machine, XGBoost, Tokenization, Stemming, Stopword Removal, Vectorization.

ABSTRACT

Sentiment analysis has become a vital technique in today's environment, greatly aiding in the comprehension of user attitudes in product reviews. This research explores sentiment analysis as it relates to reviews of E-Commerce products, primarily concentrating on polarity detection. A whole preprocessing pipeline is covered by the research, which includes actions like resolving missing values, removing symbols and punctuation, converting text to lowercase, deleting stopwords, stemming, and tokenization. After undergoing various preparation methods, the count vectorizer idea is used to convert the dataset into a numeric representation. Then, two popular machine learning techniques—Naive Bayes and Support Vector Machine—are used to determine which reviews are polarity-based and evaluate each algorithm's performance. Furthermore, an ensemble model is suggested that combines the Random Forest and XGBoost algorithms to improve polarity identification performance and accuracy even more. With the purpose of delivering a comparative analysis to support the creation of consumer purchase decisions and product enhancement plans, the study seeks to shed light on the efficacy of these algorithms in the context of E-Commerce sentiment analysis.

1. Introduction

Opinion mining, or sentiment analysis, is a computational technique that takes text data and extracts and analyzes subjective information. Sentiment analysis in the context of text-based reviews is concerned with identifying the subjective attitude or emotional tone that is expressed in a written work. Businesses, researchers, and decision-makers looking to comprehend public opinion, consumer feedback, or social media sentiments may find this to be especially helpful. There is many uses for sentiment analysis. In the business world, sentiment analysis is used by organizations to track emerging trends, measure customer satisfaction, and monitor brand perception. It is useful in social media analysis of public sentiment toward different subjects, goods, or occasions. Businesses can find opportunities to improve their products or services by analyzing the sentiments expressed in customer reviews. While negative feelings can draw attention to shortcomings, positive feelings can draw attention to strengths. Sentiment analysis helps with market research by revealing consumer attitudes and preferences regarding various goods and services. It also makes competitor analysis easier by contrasting the attitudes surrounding rival brands. The profusion of online content in the ever-changing digital age has led to an unparalleled number of user-generated reviews and comments. Product reviews are one of the most powerful of these; they are both an important source of information for customers and a vital conduit for feedback for companies. Recognizing the feelings reflected in these reviews has become essential for businesses looking to improve their offerings in line with consumer expectations. Natural language processing (NLP) contains an area called sentiment analysis that

has become quite effective at interpreting the opinions, attitudes, and emotional tones that are ingrained in textual data. This research focuses on sentiment analysis in relation to product reviews, particularly in the context of online shopping. As e-commerce platforms have grown in popularity, product reviews have emerged as a valuable resource for understanding consumer preferences, experiences, and satisfaction levels. This research main goal is to use sentiment analysis methods to determine how polarized the opinions are in the dataset of product reviews shown in figure 1. Businesses can obtain useful information about consumer satisfaction levels and areas for improvement by identifying whether feelings are favorable, negative, or neutral.

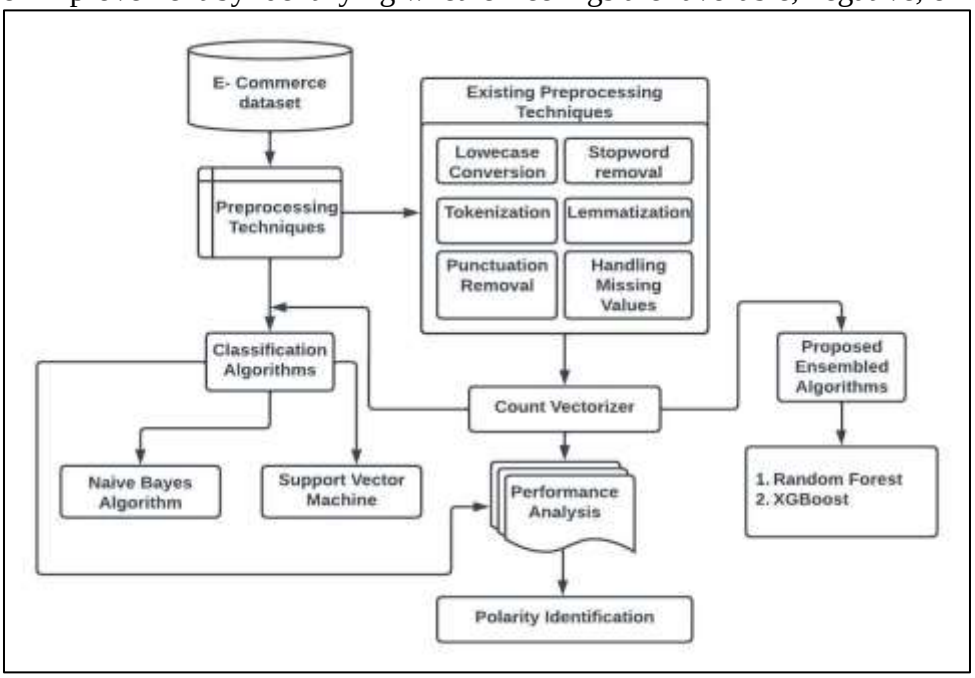

Figure 1: Workflow of Research work

To achieve proper representation of textual data, a multifaceted preparation pipeline is employed in the study. The dataset is cleaned and made ready for additional analysis through actions like converting text to lowercase, removing stopwords, stemming, tokenization, handling missing values, and getting rid of punctuation and symbols. Then, using the count vectorizer approach, the modified data is translated into a numeric representation that makes machine learning algorithms easier to use. To determine the sentiment polarity in the product reviews, this study makes use of Naive Bayes and Support Vector Machine, two popular machine learning techniques. Furthermore, an ensemble model is suggested to improve sentiment analysis performance and accuracy by fusing the Random Forest and XGBoost algorithms. The study hopes to provide insightful information about these algorithms' efficacy in the context of sentiment analysis in e-commerce through this investigation. Businesses should be able to make more informed decisions about product development, marketing tactics, and overall customer happiness thanks to the research's findings, which are predicted to provide them a deeper grasp of consumer opinion. The use of sentiment analysis in product reviews is becoming increasingly important for companies looking to gain a competitive advantage in meeting and surpassing customer expectations as the digital landscape changes.

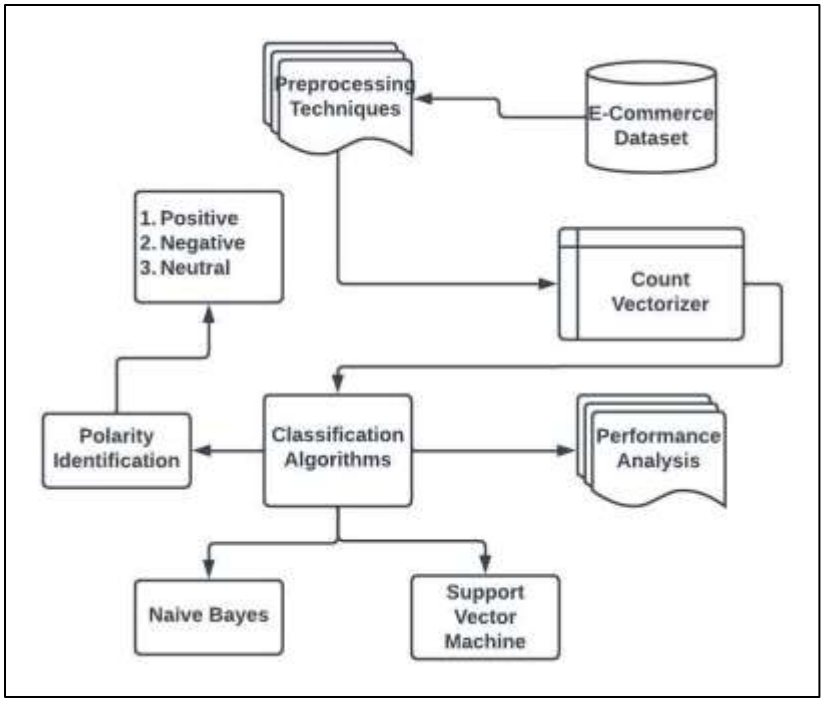

Figure 2: Architecture of Existing Algorithms

The architectures of two popular machine learning techniques, Support Vector Machine (SVM) and Naive Bayes, are used in sentiment analysis, especially when determining the polarity of customer reviews in a dataset of clothing textiles. Transform the textual information into a numerical form that machine learning algorithms can understand. TF-IDF and Bag of Words (BoW) are two popular techniques. Utilize the Bayes theorem-based Naive Bayes algorithm. It makes the assumption that a word's existence in a class is unrelated to the existence of other words. Make use of a labeled dataset to train the Naive Bayes model. Customer reviews with corresponding sentiment labels (positive, negative, or neutral) make up the dataset. Utilize the text data to extract features, like word frequencies, and feed them into the Naive Bayes classifier.

Based on the training data, the model determines the probabilities for each class (positive, negative, and neutral). The Naive Bayes model calculates the probability that a review falls into a specific sentiment category. The trained model determines the likelihood of each sentiment class for a fresh review. The anticipated sentiment for the review is assigned to the class with the highest probability. As with Naive Bayes, train the SVM model with a labeled dataset. Customer reviews with matching sentiment labels make up the dataset.

2. Review of Literature

A review of the literature aids researchers in developing a thorough grasp of the body of knowledge and research that already exists in a given area. It offers understanding of what is already known and aids in pointing out any gaps or areas that require more research. Researchers can find gaps, contradictions, or unsolved questions in the body of current knowledge by reviewing the literature. They use this to define the parameters of their own research, including its focus and scope. A literature review is an essential step in the research process because it sets the groundwork for future studies, directs the design of the study, and situates the findings within a larger academic framework.

These days, sentiment analysis is very significant. Customers' product reviews are utilized to assist consumers in making specific product purchases. Snetimental reviews aren't limited to text; they can also include emojis and audio reviews. Most text and emoji reviews are easily distinguishable by customers into three categories: neutral, negative, and positive.The following research papers

discuss how different researchers have used sentiment methods and techniques to analyze product reviews.

Zeenia Singla et al.'s study, "Sentiment Analysis of Customer Product Reviews Using Machine Learning," was covered in [10].It states that the Naïve Bayes, Support Vector Machine (SVM), and Decision Tree algorithms are used to classify the mobile phone product review provided by the customer. These techniques are applied in this study to filter and eliminate noisy data from the selected dataset, which has also been pre-processed using supervised learning to assess customers' sentimental reviews.SVM outperforms the other two methods in terms of accuracy among the three classifiers: Naïve Bayes, SVM, and Decision Tree. The study conducted in [11] by Santhosh Kumar et al. They talked about how to use algorithms like SentiWordNet, Logistic Regression, and Naïve Bayes classifier to categorize customer sentiment reviews into positive, negative, and neutral categories. After applying these algorithms to the provided dataset, the Naïve Bayes classifier emerges as the top performer among the three algorithms. Rajkumar S. et al. conducted a second study titled "Sentiment Analysis on Product Reviews Using Machine Learning Techniques" in [12]. The dataset used in this work was sourced from Amazon and comprises product reviews for televisions, tablets, smartphones, cameras, laptops, and video surveillance systems. After preprocessing, they classified reviews as positive or negative using machine learning algorithms. The benefit of machine learning techniques is concluded. Nave Bayes and Support Vector Machine yielded the best results when it came to classifying product reviews for cameras, with 98.17 percent and 93.54 percent of the reviews respectively.

A different study titled "Sentiment analysis of smart phone product review using SVM classification technique" was conducted by U Kumari et al. in [13]. In this study, sentiment reviews of smart phone products are gathered from different e-commerce sites and categorized using the Support Vector Machine (SVM) sentiment analysis algorithm. Ultimately, the SVM algorithms' performance in terms of precision, F-measures, and recall yielded the highest accuracy of 90.9%.Sentiment analysis on product reviews is a research project that Chhaya Chauhan and Smriti Sehgal discussed in [14]. This paper is primarily concerned with the extraction of customer reviews through the application of POS tagging methods and the Naïve Bayes Classifier algorithm. The study found that when compared to alternative techniques, Naive Bayes Classifiers produced good results.A study titled "Sentiment Analysis of Customer Reviews on Laptop Products for Flipkart" was conducted by Janhavi N L et al. in [15]. The researchers talked about the various techniques for grouping, categorizing, and extracting customer text-based reviews. The product reviews for toys, cameras, laptops, and cellphones are taken from Flipkart's online store. Positive, negative, and neutral texts are primarily categorized and analyzed using the CART and ROCK algorithms. Ultimately, the research project concludes that the CART algorithm yields the optimal outcome.

The study "Comparison of classification techniques for Feature Oriented Sentiment Analysis of Product Review Data" was conducted by Chetana Pujari et al. in [16]. The best way to determine the feelings that customers have expressed about the products is provided by this research work. Customer sentiment reviews are analyzed using three classification algorithms: Naïve Bayes, SVM, and Maximum Entropy. After comparing the algorithms' performances, it is determined that the Naïve Bayes algorithm offers the highest accuracy. A study conducted in [17] by Shivaprasad T K and Jyothi Shetty. The article goes on to explain that sentiment analysis is a field of study that uses natural language processing (NLP) to analyze and extract opinions from a given review. This study presents the taxonomy of several sentiment analysis techniques. This study shows that Support Vector Machine (SVM) offers superior accuracy when compared to Nave Bayes and maximum entropy approaches. Subhabrata Mukherjee and Pushpak Bhattacharyya's study, "Feature Specific Sentiment Analysis for Product Reviews," was published in [18]. They created a system for classifying opinion expressions that suggest possible features in reviews in this paper. At last, the user-specified feature's opinion expression is retrieved by the system. The system

performed better than the baseline systems in every evaluation across a range of dimensions and two datasets. The system significantly outperformed both the more sophisticated and the naïve baselines.

Xing Fang and Justin Zhan's study, "Sentiment analysis using product review data," was published in [19]. This study aims to tackle sentiment polarity categorization, one of the most basic problems in sentiment analysis. Comprehensive process descriptions are provided, along with a generic process for sentiment polarity classification. Online product reviews on Amazon.com provided the study's data. In this study, the Naive Bayes classifier, SVM, and Random classifier methods were used to classify the data. The Random Forest method yields the best results when compared to other methods, according to the research findings. The study conducted by Aytug Onan in [20] is titled "Sentiment analysis on product reviews based on weighted word embeddings and deep neural networks." Sentiment analysis is a crucial task in natural language processing that entails obtaining attitudes, thoughts, opinions, or judgments regarding a particular subject. To separate the weighted and unweighted words from the text-based data, a number of weighting functions are used, including IDF, TLF, and SIF. Ultimately, they came to the conclusion that the suggested algorithms offer the best accuracy of 93.85%.

Comparative Study of Sentiment Analysis with Product Reviews Using Machine Learning and Lexicon-Based Approaches is the title of another study conducted by Nguye et al. in [21]. This study looked at how the lexicon-based techniques of valence aware dictionary, sentiwordnet, and sentiment reasoner pattern, as well as the machine learning algorithms of logistic regression, support vector machine, and boosting, are applied to customer reviews gathered from the Amazon shopping website. In the end, they come to the conclusion that, when compared to other approaches, machine learning algorithms produced the best accuracy.

Table 1: Comparison of various research methods and results

Paper Ref. No.	Methods Used	Results & Accuracy
10	Naïve Bayes, Support Vector Machine, Decision Tree	SVM yields the best method of 81.75%.
11	Logistic Expression, SentiWordNet and Naïve Bayes,	Naïve Bayes algorithm outperforms the other algorithms
12	Naïve Bayes , Support Vector Machine.	Naïve Bayes algorithm provides 98.17 % accuracy
13	Support Vector Machine	Support Vector Machine algorithm yields 90.99 % of highest accuracy.
14	Naïve Bayes Algorithm and POS tagging method.	Naïve Bayes outperforms the other algorithms and yields the highest percentage of accuracy
15	CART and ROCK algorithms	CART provides the best accuracy.
16	Naïve Bayes, Support Vector Machine, Maximum Entropy classifier.	Support Vector Machine has yields 82.85% of accuracy compared to others.
17	Naïve Bayes, Support Vector Machine, Maximum Entropy.	Support Vector Machine yields the highest percentage of accuracy
18	Support Vector Machine, Naïve Base line Proposed algorithm.	Naïve Base line algorithms provides the best result.
19	Naive Bayes classifier, Support Vector Machine, and Random classifier	Random Forest method provides the best result

20	CNN, RNN, LSTM, GRU	CNN-LSTM Proposed architecture obtained the highest of 93.85% accuracy.
21	Machine Learning Algorithm (Logistic Regression, Support Vector Machine, Boosting), Lexicon based approach (valence aware dictionary, sentiwordnet, Sentiment reasoner pattern)	Machine Learning algorithms provide the best accuracy.

This review of literature can help the researchers to find creative and novel methods that have worked well in related studies by looking through the literature. Understanding the difficulties others have faced can help with method selection and implementation. Literature reviews frequently offer information on practical considerations related to the application of specific methods, such as data collection, sample sizes, ethical considerations, and other practical aspects of research. Researchers can learn from both successful and unsuccessful applications of methods by studying the literature.

3. Description of Dataset

The results of the preprocessing techniques have been analyzed in this research study. For this research project, the data set was obtained from the Kaggle repository. Pre-processing techniques or further investigation were applied to the real dataset in order to classify it for text-based analysis. There are 300 records in the dataset for electronic product reviews. the same to be subjected to Python programming analysis and preprocessing.

Table 2: Description of Attributes

Attribute Name	***Description***
Unnamed: 0	This column seems to be an unnamed or default index column. It often represents the row number or a unique identifier for each record.
Clothing ID	This attribute appears to be an identifier for the clothing items in the dataset. It might be a unique identifier for each piece of clothing.
Age	This column likely represents the age of the individuals providing reviews or feedback on the clothing items.
Title	This attribute may contain the title or headline of the reviews given by customers for the clothing items.
Review Text	This column probably contains the main body of text where customers provide detailed reviews and feedback on the clothing items.
Rating	This attribute likely contains numerical values representing the ratings given by customers for the clothing items. Ratings are often on a scale, such as 1 to 5.
Recommended IND	This column may contain binary values indicating whether the customer recommended the clothing item or not. It could be a binary indicator (0 or 1).
Positive Feedback Count	This attribute seems to represent the count of positive feedback received for a particular clothing item or review.
Division Name	This column may contain the name or category of the division to which the clothing item belongs.
Department Name	This attribute likely contains the name or category of the department to which the clothing item belongs.
Class Name	This column may contain the class or category name of the clothing item.

Table 2 shows the number of attributes in the chosen dataset and the target attribute from the above table is “Review Text”. This attribute contains the text-based reviews which are given by the

customer. That particular attribute is chosen for further analysis to predict the polarity from the text. Table 3 shows the sample dataset which contains the reviews and all the other data's related to the attribute.

Table 3: Sample Dataset

Unnamed: 0	0	1	2	3	4	5	6	7	8
Clothing ID	767	1080	1077	1049	847	1080	858	858	1077
Age	33	34	60	50	47	49	39	39	24
Title			Some	My	Flattering	Not for	Cagrcoal	Shimmer,	Flattering
Review Text	Absolutely	Love this dress! it's	I had such high	I love, love, love	This shirt is very	I love tracy reese	I aded this in my	I ordered this in	I love this dress. i
Rating	4	5	3	5	5	2	5	4	4
Recommended IND	1	1	0	1	1	0	1	1	0
Positive Feedback	0	4	0	0	6	4	1	4	0
Division Name	Initmates	General	General	General	General	General	General	General	General
Department Name	Intimate	Dresses	Dresses	Bottoms	Tops	Dresses	Tops	Tops	Dresses
Class Name	Intimates	Dresses	Dresses	Pants	Blouses	Dresses	Knits	Knits	Dresses

4. Preprocessing Techniques

Preprocessing is often necessary for text data before sentiment analysis can be used. To set up the text for analysis, this involves chores like stop word removal, tokenization, and stemming. An essential phase in the pipeline for data analysis and machine learning is preprocessing. Preprocessing eliminates unnecessary or redundant data, which increases model training efficiency. When working with large datasets or computationally demanding algorithms, this is especially crucial. In order to prepare raw data for analysis or model training, it must be cleaned and transformed. These preprocessing methods are not all-inclusive, and the methods selected will rely on the particulars of the data as well as the demands of the machine learning or analysis task. It's common practice to try out various preprocessing steps in order to determine which combination works best for a particular dataset and problem. preprocessing methods are essential for making sure data is prepared for modeling and analysis. They solve problems with data quality, enhance model performance, and make it easier to extract valuable insights from large, complicated datasets, all of which contribute to the overall success of machine learning and data analytics research work.Certain preprocessing procedures that are suited to the features of the data may be needed for different domains. It is essential to comprehend the problem and the domain in order to choose the best preprocessing methods.

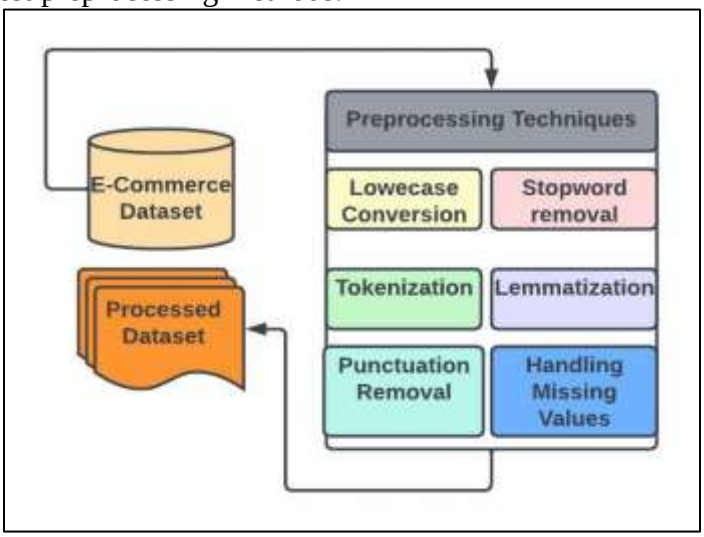

Figure 3: Preprocessing Techniques

Managing Missing Data Imputation and other more sophisticated approaches, such as mean, median, and mode, can be used to impute missing values. Eliminate any rows or columns that lack values if they don't hold important information. Finding and eliminating outliers that could have a negative impact on the analysis or model performance is known as outlier detection and removal. Noise reduction is the process of reducing noise in data by clustering, filtering, or binning. To guarantee that the ranges of numerical features are comparable, standardize them to a standard scale (such as z-score normalization).Categorical Data Encoding generate binary vectors from categorical variables. Transform numerical values into categorical labels using label encoding.

Lower the text: One of the most common text preparation techniques in Python is to convert the text to the same case, preferably lower case. However, since lower casing can sometimes cause information loss, you are not required to finish this step every time you work on a nlp problem. Phrases written in upper case, for example, can convey discontent with how one handles one's emotions in any endeavor.

Before Lowering the Text

Absolutely wonderful - silky and sexy and comfortable

After Lowering the Text

Absolutely wonderful - silky and sexy and comfortable

Table 4: Comparison of before and after preprocessing – Lowercase conversion

Step 0 - Original	Step 1 - Lowercase
Absolutely wonderful - silky and sexy and comfortable	absolutely wonderful - silky and sexy and comfortable
Love this dress! it's sooo pretty. i happened to find it in a store, and i'm glad i did bc i never would have ordered it online bc it's petite. i bought a petite and am 5'8". i love the length on me- hits just a little below the knee. would definitely be a true midi on someone who is truly petite.	love this dress! it's sooo pretty. i happened to find it in a store, and i'm glad i did bc i never would have ordered it online bc it's petite. i bought a petite and am 5'8". i love the length on me- hits just a little below the knee. would definitely be a true midi on someone who is truly petite.
Dress runs small esp where the zipper area runs. i ordered the sp which typically fits me and it was very tight! the material on the top looks and feels very cheap that even just pulling on it will cause it to rip the fabric. pretty disappointed as it was going to be my christmas dress this year! needless to say it will be going back.	dress runs small esp where the zipper area runs. i ordered the sp which typically fits me and it was very tight! the material on the top looks and feels very cheap that even just pulling on it will cause it to rip the fabric. pretty disappointed as it was going to be my christmas dress this year! needless to say it will be going back.

Tokenization: In natural language processing (NLP), tokenization is a critical stage in which text is broken down into smaller units known as tokens. Word tokenization divides the text into discrete words, or tokens. Ideal for tasks like text classification, sentiment analysis, or word frequency analysis where word-level analysis is required.

Before Tokenizing the Text

This Dress is Perfection! So Pretty And Flattering.

After Tokenizing the Text

['this', 'dress', 'is', 'perfection', '!', 'so', 'pretty', 'and', 'flattering', '.']

Table 5: Comparison of before and after preprocessing – Tokenization

Step 0 - Original	Step 2 - Tokenization
This shirt is very flattering to all due to the adjustable front tie. it is the perfect length to wear with leggings and it is sleeveless so it pairs well with any cardigan. love this shirt!!!	['this', 'shirt', 'is', 'very', 'flattering', 'to', 'all', 'due', 'to', 'the', 'adjustable', 'front', 'tie', '.', 'it', 'is', 'the', 'perfect', 'length', 'to', 'wear', 'with', 'leggings', 'and', 'it', 'is', 'sleeveless', 'so', 'it', 'pairs', 'well', 'with', 'any', 'cardigan', '.', 'love', 'this', 'shirt', '!', '!', '!']
I aded this in my basket at hte last mintue to see what it would look like in person. (store pick up). i went with teh darkler color only because i am so pale :-) hte color is really gorgeous, and turns out it mathced everythiing i was trying on with it prefectly. it is a little baggy on me and hte xs is hte msallet size (bummer, no petite). i decided to jkeep it though, because as i said, it matvehd everything. my ejans, pants, and the 3 skirts i waas trying on (of which i]kept all) oops.	['i', 'aded', 'this', 'in', 'my', 'basket', 'at', 'hte', 'last', 'mintue', 'to', 'see', 'what', 'it', 'would', 'look', 'like', 'in', 'person', '.', '(', 'store', 'pick', 'up', ')', '.', 'i', 'went', 'with', 'teh', 'darkler', 'color', 'only', 'because', 'i', 'am', 'so', 'pale', ':', '-', ')', 'hte', 'color', 'is', 'really', 'gorgeous', ',', 'and', 'turns', 'out', 'it', 'mathced', 'everythiing', 'i', 'was', 'trying', 'on', 'with', 'it', 'prefectly', '.', 'it', 'is', 'a', 'little', 'baggy', 'on', 'me', 'and', 'hte', 'xs', 'is', 'hte', 'msallet', 'size', '(', 'bummer', ',', 'no', 'petite', ')', '.', 'i', 'decided', 'to', 'jkeep', 'it', 'though', ',', 'because', 'as', 'i', 'said', ',', 'it', 'matvehd', 'everything', '.', 'my', 'ejans', ',', 'pants', ',', 'and', 'the', '3', 'skirts', 'i', 'waas', 'trying', 'on', '(', 'of', 'which', 'i', ']', 'kept', 'all', ')', 'oops', '.']

Removal of Punctuation: In many natural language processing (NLP) tasks, preprocessing involves removing symbols and punctuation from text. Usually, the goal is to make the text simpler and concentrate only on the words, omitting special characters and punctuation.

Before Removal of Punctuation Example

Love this dress! it's sooo pretty. i happened to find it in a store, and i'm glad i did bc i never would have ordered it online bc it's petite. i bought a petite and am 5'8". i love the length on me- hits just a little below the knee. would definitely be a true midi on someone who is truly petite.

After Removal of Punctuation Example

['love', 'this', 'dress', 'it', 'sooo', 'pretty', 'i', 'happened', 'to', 'find', 'it', 'in', 'a', 'store', 'and', 'i', 'glad', 'i', 'did', 'bc', 'i', 'never', 'would', 'have', 'ordered', 'it', 'online', 'bc', 'it', 'petite', 'i', 'bought', 'a', 'petite', 'and', 'am', '5', '8', 'i', 'love', 'the', 'length', 'on', 'hits', 'just', 'a', 'little', 'below', 'the', 'knee', 'would', 'definitely', 'be', 'a', 'true', 'midi', 'on', 'someone', 'who', 'is', 'truly', 'petite']

Table 6: Results of Punctuation Removal

Step 0 - Original	Step 3- Remove Punctuation
This dress is perfection! so pretty and flattering.	['this', 'dress', 'is', 'perfection', 'so', 'pretty', 'and', 'flattering']
More and more i find myself reliant on the reviews written by savvy shoppers before me and for the most past, they are right on in their estimation of the product. in the case of this dress-if it had not been for the reveiws-i doubt i would have even tried this. the dress is beautifully made, lined and reminiscent of the old retailer quality. it is lined in the solid periwinkle-colored fabric that matches the outer fabric print. tts and very form-fitting. falls just above the knee and does not rid	['more', 'and', 'more', 'i', 'find', 'myself', 'reliant', 'on', 'the', 'reviews', 'written', 'by', 'savvy', 'shoppers', 'before', 'me', 'and', 'for', 'the', 'most', 'past', 'they', 'are', 'right', 'on', 'in', 'their', 'estimation', 'of', 'the', 'product', 'in', 'the', 'case', 'of', 'this', 'it', 'had', 'not', 'been', 'for', 'the', 'doubt', 'i', 'would', 'have', 'even', 'tried', 'this', 'the', 'dress', 'is', 'beautifully', 'made', 'lined', 'and', 'reminiscent', 'of', 'the', 'old', 'retailer', 'quality', 'it', 'is', 'lined', 'in', 'the', 'solid', 'fabric', 'that', 'matches', 'the', 'outer', 'fabric', 'print', 'tts', 'and', 'very', 'falls', 'just', 'above', 'the', 'knee', 'and', 'does', 'not', 'rid']

Stop word Removal: A typical preprocessing step in text analysis and natural language processing (NLP) is stopword removal. Words that are commonly used in a language but usually add little to the meaning of a text are known as stopwords. English stopwords include "the," "is," "and," "of," and so forth. Eliminating stopwords from text reviews can help the reader concentrate on the more significant words that convey the message or crucial details.

Before Stop Word Removal Example:

I love, love, love this jumpsuit. it's fun, flirty, and fabulous! every time i wear it, i get nothing but great compliments!

After Stop Word Removal Example:

['love', 'love', 'love', 'jumpsuit', 'fun', 'flirty', 'fabulous', 'every', 'time', 'wear', 'get', 'nothing', 'great', 'compliment']

Table 7: Results of Stopword Removal

Step 0 - Original	Step 4 - Remove Stopwords
Material and color is nice. the leg opening is very large. i am 5'1 (100#) and the length hits me right above my ankle. with a leg opening the size of my waist and hem line above my ankle, and front pleats to make me fluffy, i think you can imagine that it is not a flattering look. if you are at least average height or taller, this may look good on you.	['material', 'color', 'nice', 'leg', 'opening', 'large', '5', '1', '100', 'length', 'hits', 'right', 'ankle', 'leg', 'opening', 'size', 'waist', 'hem', 'line', 'ankle', 'front', 'pleats', 'make', 'fluffy', 'think', 'imagine', 'flattering', 'look', 'least', 'average', 'height', 'taller', 'may', 'look', 'good']
Took a chance on this blouse and so glad i did. i wasn't crazy about how the blouse is photographed on the model. i paired it whit white pants and it worked perfectly. crisp and clean is how i would describe it. launders well. fits great. drape is perfect. wear tucked in or out - can't go wrong.	['took', 'chance', 'blouse', 'glad', 'crazy', 'blouse', 'photographed', 'model', 'paired', 'whit', 'white', 'pants', 'worked', 'perfectly', 'crisp', 'clean', 'would', 'describe', 'launders', 'well', 'fits', 'great', 'drape', 'perfect', 'wear', 'tucked', 'ca', 'go', 'wrong']
A flattering, super cozy coat. will work well for cold, dry days and will look good with jeans or a dressier outfit. i am 5' 5", about 135 and the small fits great.	['flattering', 'super', 'cozy', 'coat', 'work', 'well', 'cold', 'dry', 'days', 'look', 'good', 'jeans', 'dressier', 'outfit', '5', '5', '135', 'small', 'fits', 'great']

Lemmatization

The process of reducing words to their lemma, or base or root form, is called lemmatization. Lemmatization is frequently used in text-based reviews to standardize vocabulary and condense inflected words to a common base form. This reduces word variation into a single representation, which aids in text analysis, sentiment analysis, and other natural language processing tasks.

Before Lemmatization Example:

I love this dress. i usually get an xs but it runs a little snug in bust so i ordered up a size. very flattering and feminine with the usual retailer flair for style.

After Lemmatization Example:

['love', 'dress', 'usually', 'get', 'x', 'run', 'little', 'snug', 'bust', 'ordered', 'size', 'flattering', 'feminine', 'usual', 'retailer', 'flair', 'style']

Table 8: Results of Lemmatization

Step 0 - Original	Step 5 – Lemmatization
Cute little dress fits tts. it is a little high waisted. good length for my 5'9 height. i like the dress, i'm just not in love with it. i dont think it looks or feels cheap. it appears just as pictured.	['cute', 'little', 'dress', 'fit', 'tt', 'little', 'high', 'waisted', 'good', 'length', '5', '9', 'height', 'like', 'dress', 'love', 'dont', 'think', 'look', 'feel', 'cheap', 'appears', 'pictured']
I love this shirt because when i first saw it, i wasn't sure if it was a shirt or dress. since it is see-through if you wear it like a dress you will need a slip or wear it with leggings. i bought a slip, wore the tie in the back, and rocked it with white wedges. you could also wear it as a vest. be careful with the buttons. i haven't had any fall off yet, but i feel like they will. overall it's great for any occasion and it's fun to wear!	['love', 'shirt', 'first', 'saw', 'sure', 'shirt', 'dress', 'since', 'wear', 'like', 'dress', 'need', 'slip', 'wear', 'legging', 'bought', 'slip', 'wore', 'tie', 'back', 'rocked', 'white', 'wedge', 'could', 'also', 'wear', 'vest', 'careful', 'button', 'fall', 'yet', 'feel', 'like', 'overall', 'great', 'occasion', 'fun', 'wear']

Handling Missing Value

Finding the reviews with missing values should be your first step. This step assists you in determining the scope of the problem and how it affects your dataset. Should the quantity of reviews containing missing values be minimal and not substantially impact the dataset, you might decide to eliminate those occurrences. Although it's common practice to use 'nan' as a placeholder, it's important to remember that models or downstream analyses should be built to handle or exclude these placeholder values appropriately.Examine the text length distribution in the reviews that aren't empty, and then use statistical measures like the mean or median to fill in the blanks for the reviews that aren't empty.

Table 9: Handling Missing Values

Step 0 - Original	Step 1 - Lowercase	Step 2 - Tokenization	Step 3 - Remove Punctuation	Step 4 - Remove Stopwords	Step 5 - Lemmatization	Step 6 - After Missing Values Handling
		['nan']	['nan']	['nan']	['nan']	['nan']
		['nan']	['nan']	['nan']	['nan']	['nan']
		['nan']	['nan']	['nan']	['nan']	['nan']
		['nan']	['nan']	['nan']	['nan']	['nan']
		['nan']	['nan']	['nan']	['nan']	['nan']
		['nan']	['nan']	['nan']	['nan']	['nan']
		['nan']	['nan']	['nan']	['nan']	['nan']
		['nan']	['nan']	['nan']	['nan']	['nan']
		['nan']	['nan']	['nan']	['nan']	['nan']
		['nan']	['nan']	['nan']	['nan']	['nan']
		['nan']	['nan']	['nan']	['nan']	['nan']

Using the features of previos reviews as a guide, this technique creates artificial content. The string "nan" is used to fill in the missing values in the "customer_review" column using fillna("nan").By setting inplace=True, that can be sure that the modifications take effect immediately on the original DataFrame .

Example: Missing values filled with ['nan’] in the text review

5. Materials and methods

Naïve Bayes: Based on the "naive" assumption of feature independence, the Bayes algorithm is a probabilistic classification method. In spite of its simplicity, it frequently works well in practice and is especially helpful for spam filtering and text classification. Using the training dataset as a basis, determine the prior probability of each class.Determine the probability of each feature for every class.This entails normalizing and counting the instances of each feature within each class.select the class with the highest posterior probability as the predicted class

```
# Naïve Bayes Pseudo code for Training
for each class C_i:
   calculate prior probability P(C_i)
   for each feature x_j:
      calculate likelihood P(x_j | C_i)

# Prediction
for each class C_i:
   calculate posterior probability P(C_i | x_1, x_2, ..., x_n)
```

Given the class, the algorithm assumes that the features are conditionally independent. Even though real-world data frequently contradicts this assumption, the algorithm can still function surprisingly well in practice.The specifics of the actual implementation may differ, particularly in terms of smoothing methods for handling zero probabilities and other optimizations.

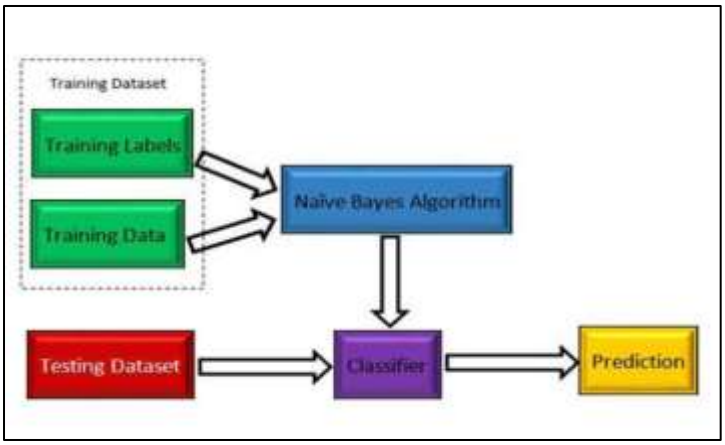

Figure 4: Naïve Bayes Work flow

To prevent zero probabilities, smoothing techniques such as Laplace smoothing are frequently employed. A high-level understanding of the steps in the Naive Bayes algorithm can be obtained from the pseudocode above.

Support Vector Machine: Encouragement A potent supervised machine learning algorithm for regression and classification applications is called Vector Machine. The process involves identifying the hyperplane that divides the data into the most distinct classes.The SVM algorithm looks for a hyperplane that maximizes the margin between distinct classes given a set of labeled training data. The hyperplane that maximally separates the support vectors—the data points nearest to the decision boundary—is chosen as the chosen one. To convert the input features into a higher-dimensional space, SVM can make use of a kernel function. The algorithm can now locate a nonlinear decision boundary as a result.

 # Pseudo Code for support Vector Machine Training

given training data (X_train, y_train):
    choose a kernel function (e.g., linear, polynomial, radial basis function)
    initialize parameters (e.g., C for regularization, kernel parameters)
    compute the kernel matrix K(X_train, X_train)
    set up optimization to find alpha_i values using the support vector machine dual problem
    solve the optimization problem to obtain alpha_i values
    compute the weight vector and bias term from the alpha_i values and support vectors
Prediction
for a new data point X_test:
    compute the kernel values between X_test and the support vectors
predict the class by evaluating the decision function using the weight vector and bias term

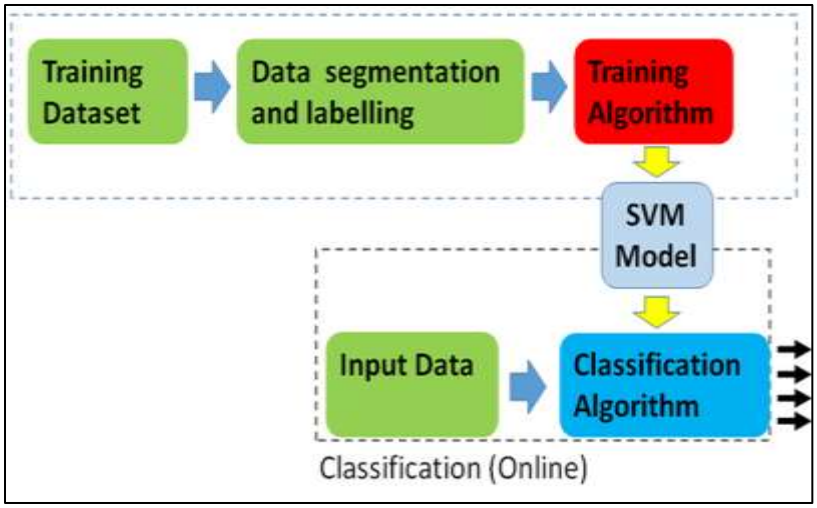

Figure 5: Workflow of Support Vector Machine

The distance between the closest data point from either class and the hyperplane is known as the margin. Increasing the margin contributes to the model's improved generalization. The data points that are closest to the decision boundary are known as support vectors, and they are essential in helping to define the hyperplane. Non-linearly separable data can be handled by adapting SVM to allow for some misclassifications. A regularization parameter (C) regulates the degree of tolerance, and a soft margin is used to accomplish this.

Ensemble Method: In addition to improving accuracy, robustness, and generalization, an ensemble method combining Random Forest and XGBoost for sentiment analysis often outperforms individual models by leveraging multiple models for prediction; comparing the ensemble method's performance to that of Naive Bayes (NB) and Support Vector Machine (SVM) offers valuable information about the advantages and disadvantages of each approach. Utilizing the sentiment analysis dataset, train distinct Random Forest and XGBoost models. These models are capable of capturing non-linear patterns in the data and managing intricate relationships. Mix the Random Forest and XGBoost predictions together using an appropriate ensemble method, like voting or averaging. By utilizing the advantages of each model individually, this ensemble approach generates predictions that are more reliable and accurate. Make sure the models are diverse by adjusting the hyperparameters for Random Forest and XGBoost in different ways. This enhances the ensemble's overall performance and helps avoid overfitting. Use cross-validation or a validation set to assess the ensemble method. Standard metrics like accuracy, precision, recall, and F1 score are used to measure performance.

Utilizing the same metrics (accuracy, precision, recall, and F1 score) as the ensemble method, compare the performance of NB and SVM. Examine and contrast the results of the ensemble approach, SVM, and NB. Determine which method for the sentiment analysis task produces the best results in terms of accuracy and generalization. Analyze each method's computational complexity. Compared to NB and SVM, ensemble methods may require more processing power, particularly when dealing with multiple models. Think about the compromise between computational cost and performance.

6. Experimental Result

The experiment's objective, such as sentiment analysis or text classification, and the text-based dataset's significance in answering the study question. Identify the performance of the text-based datasetby using some preprocessing techniques using Tokenization, stemming, and any that were used on the text data. Draw attention to any difficulties while preparing the data. After the preprocessing steps, the text-based data is converted into numeric form for further analysis. The

machine learning models used, with a particular emphasis on Support Vector Machine and Naive Bayes and also the proposed method was used to analyze the text based data to identify the polarity of the text based reviewsto help the business and foer the customer to purchase the particular product. Give a succinct justification for the models chosen for text-based analysis. Finally, the evaluation metrics that were used, including ROC-AUC, F1 score, accuracy, precision, and recall.

6.1 Result of processed data using Naïve Bayes, SVMand Proposed Method

In this research, the perfrmance of four machine learning algorithms (Naive Bayes, SVM, Random Forest, and Decision Tree) for sentiment analysis using processed data. The dataset was pre-processed by removing stopwords, performing tokenization, and applying lemmatization. The evaluation metrics used for comparing the models were precision, recall, and f1-score for each sentiment class, as well as the accuracy, macro average, and weighted average across all sentiment classes.The results show that all models performed better with the processed data than with the raw data. Among the four models, Random Forest performed the best with an accuracy of 0.89, a weighted average f1-score of 0.87, and a weighted average precision of 0.88. It outperformed the other models in all sentiment classes except for Neutral, where it had the same performance as SVM.

SVM also performed well with an accuracy of 0.86, a weighted average f1-score of 0.83, and a weighted average precision of 0.81. It had the best precision for the Negative sentiment class among all models. Naive Bayes had an accuracy of 0.86, a weighted average f1-score of 0.81, and a weighted average precision of 0.81.Among these models, the proposed ensemble method was the best performing model for sentiment analysis on this dataset.

Table 10: Performance metrics for Naive Bayes model using pre-processed data

Naïve Bayes	**Precision**	**Recall**	**f1-Score**
Negative	0.80	0.22	0.35
Neutral	0.00	0.00	0.00
Positive	0.86	0.99	0.92
Accuracy			0.86
Macro Average	0.55	0.4	0.42
Weighted Average	0.81	0.86	0.81

In Table 10, With an F1-score of 0.92, the model has reasonably high precision and recall for the positive emotion class when compared to other sentiment classes. The model's lower F1-score of 0.35 for the negative sentiment class shows that it is not performing as well for negative sentiment.For the neutral emotion class, the model has a perfect precision of 1.00, but its recall is zero, giving it a comparatively low F1-score of 0.00. It can be inferred from this that the model is unable to recognise neutral sentiment.

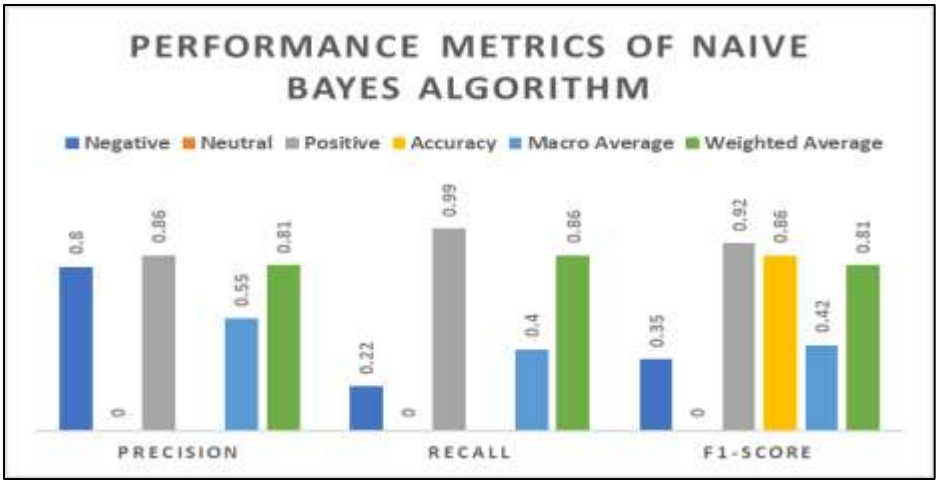

Figure 6: Performance Metrics of processed data using Naïve Bayes Model

Figure 8 shows that the model's overall accuracy is 0.86, meaning that 86% of all cases were properly identified. The macro average F1-score for all sentiment classes is 0.42, which is a comparatively poor indicator of overall success. As indicated by the weighted average F1-score of 0.81, the model appears to be working better for the majority class.

Table 11: Performance Metrics of Support Vector Machine model using preprocessed data

SVM	Precision	Recall	f1-Score
Negative	0.71	0.33	0.45
Neutral	0.00	0.00	0.00
Positive	0.87	0.98	0.92
Accuracy			0.86
Macro Average	0.53	0.44	0.46
Weighted Average	0.81	0.86	0.83

Table 11 shows with an F1-score of 0.92, the model has reasonably high precision and recall for the positive emotion class when compared to other sentiment classes. The model's lower F1-score of 0.45 for the negative sentiment class shows that it is not performing as well for negative sentiment. For the neutral emotion class, the model has a perfect precision of 1.00, but its recall is zero, giving it a comparatively low F1-score of 0.00. It can be inferred from this that the model is unable to recognise neutral sentiment.

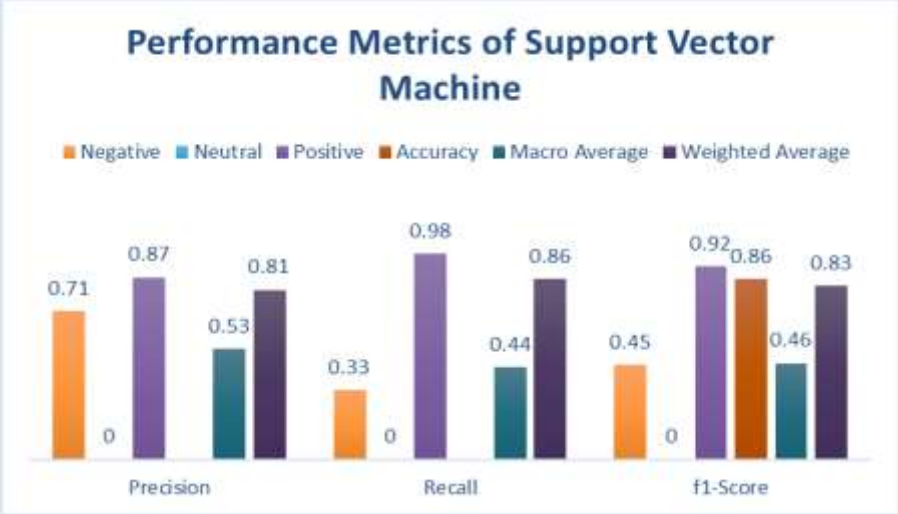

Figure 7: Classification Report of processed data using Support Vector Machine

Figure 7 shows that the model's overall accuracy is 0.86, meaning that 86% of all cases were properly identified. The macro average F1-score for all sentiment classes is 0.46, which is a comparatively poor indicator of overall performance. According to the weighted average F1-score of 0.83, the model appears to be working better for the majority class.

Table 12: Performance Metrics of Ensemble model

Random Forest	Precision	Recall	f1-Score
Negative	0.68	0.47	0.56
Neutral	1.00	0.31	0.47
Positive	0.9	0.98	0.94
Accuracy			0.89
Macro Average	0.86	0.59	0.66
Weighted Average	0.88	0.89	0.87

Table 12 shows that examining the precision, recall, and F1-score for each sentiment class, the model has a high F1-score of 0.94 and high precision and recall for the positive sentiment class. The model's comparatively lower F1-score of 0.56 for the negative sentiment class shows that it is not performing as well for negative sentiment.

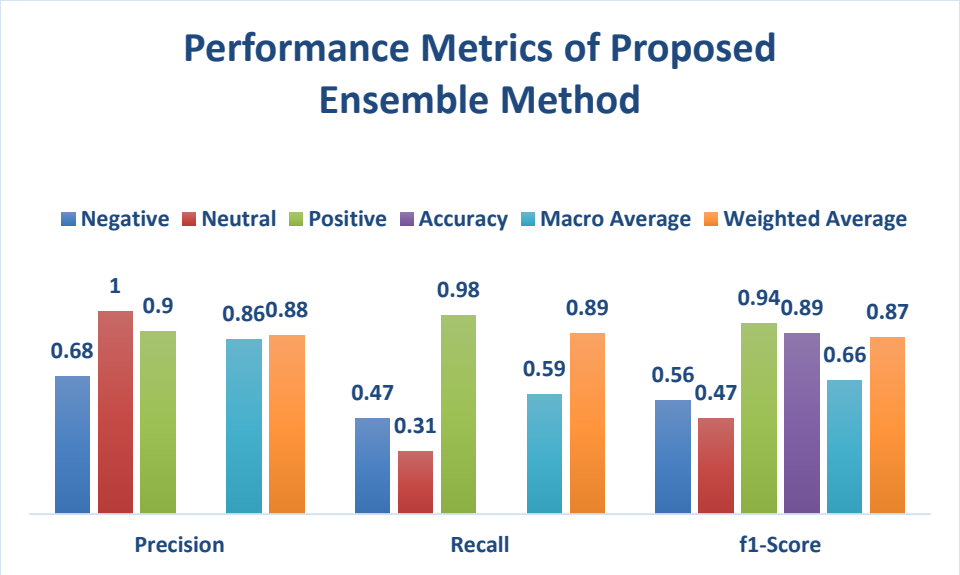

Figure 8: Performance Metrics of processed data using Ensemble Method

Figure 8 shows that the model's precision for the neutral sentiment class is a flawless 1.00, but its recall is only 0.31, giving it an F1-score of 0.47. This indicates that the model is ineffective in detecting neutral sentiment. The model's overall accuracy is 0.89, meaning that 89% of all cases were properly identified. The macro average F1-score, which measures performance across all sentiment classes on average, is 0.66, which is a respectable value. As indicated by the weighted average F1-score of 0.87, the model appears to be working better for the majority class.

Table 13: Number of words present in the Reviews before preprocessing

Reviews	Number of words in review
Bought the black xs to go under the larkspur midi dress because they didn't bother lining the skirt portion (grrrrrrrrrr). my stats are 34a-28/29-36 and the xs fit very smoothly around the chest and was flowy around my lower half, so i would say it's running big. the straps are very pretty and it could easily be nightwear too. i'm 5'6" and it came to just below my knees.	377
This is a nice choice for holiday gatherings. i like that the length grazes the knee so it is conservative enough for office related gatherings. the size small fit me well - i am usually a size 2/4 with a small bust. in my opinion it runs small and those with larger busts will definitely have to size up (but then perhaps the waist will be too big). the problem with this dress is the quality. the fabrics are terrible. the delicate netting type fabric on the top layer of skirt got stuck in the zip	500
I took these out of the package and wanted them to fit so badly, but i could tell before i put them on that they wouldn't. these are for an hour-glass figure. i am more straight up and down. the waist was way too small for my body shape and even if i sized up, i could tell they would still be tight in the waist and too roomy in the hips - for me. that said, they are really nice. sturdy, linen-like fabric, pretty color, well made. i hope they make someone very happy!	470

Material and color is nice. the leg opening is very large. i am 5'1 (100#) and the length hits me right above my ankle. with a leg opening the size of my waist and hem line above my ankle, and front pleats to make me fluffy, i think you can imagine that it is not a flattering look. if you are at least average height or taller, this may look good on you.	359
Took a chance on this blouse and so glad i did. i wasn't crazy about how the blouse is photographed on the model. i paired it whit white pants and it worked perfectly. crisp and clean is how i would describe it. launders well. fits great. drape is perfect. wear tucked in or out - can't go wrong.	296
A flattering, super cozy coat. will work well for cold, dry days and will look good with jeans or a dressier outfit. i am 5' 5", about 135 and the small fits great.	167
I love the look and feel of this tulle dress. i was looking for something different, but not over the top for new year's eve. i'm small chested and the top of this dress is form fitting for a flattering look. once i steamed the tulle, it was perfect! i ordered an xsp. length was perfect too.	292

The initial length or word count of each review is shown in this raw data. Text data may have a number of components before preprocessing, such as stop words, punctuation, and possibly unnecessary or redundant information shown in table 13.The different word counts represent the different lengths of the reviews. While some reviews are shorter (167 words), others are longer (500 words, for example). Review length variability is a common feature of natural language data. Longer reviews might be more detailed, but they might also be filled with extraneous information, irrelevant details, or noise. By eliminating extraneous information and concentrating on the most important parts of the text, preprocessing techniques can aid in text cleanup.

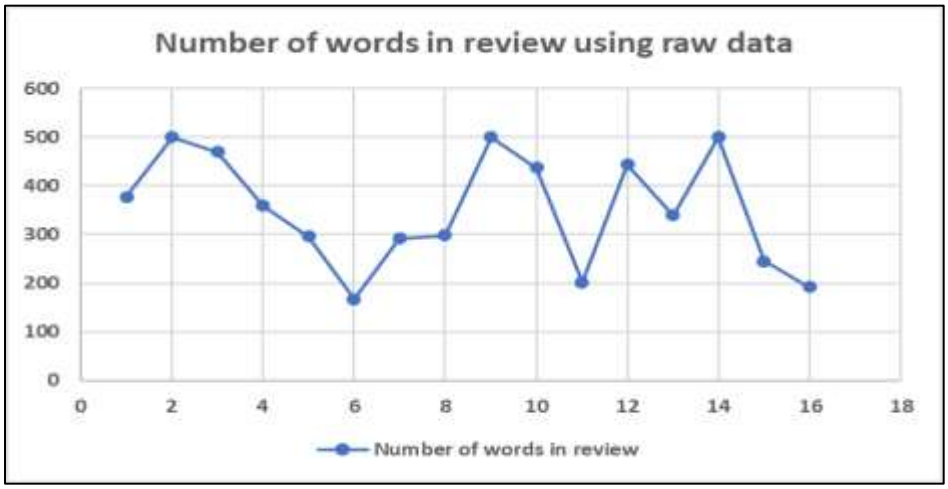

Figure 9: Number of words present in the Reviews before preprocessing

Figure 9 shows that the preprocessing methods like tokenization, stop word removal, stemming, and lemmatization can be used, given the initial word counts, to improve the text's quality and streamline it for further analysis. While shorter reviews might offer succinct opinions, longer reviews might indicate more complex or detailed feedback. The particular objectives of the analysis and the properties of the dataset will determine which preprocessing methods are used. Larger datasets from longer reviews may have an impact on the amount of computing power needed for processing, storing, and analyzing the data. Preprocessing facilitates more effective

resource management. An initial examination of the unprocessed data sheds light on the distribution of review durations. This knowledge can help determine how much preprocessing is necessary to get the desired results.

Table 14: Number of words present in the Reviews after preprocessing

Reviews	Number of words in review
['bought', 'black', 'x', 'go', 'larkspur', 'midi', 'dress', 'bother', 'lining', 'skirt', 'portion', 'grrrrrrrrrrr', 'stats', 'x', 'fit', 'smoothly', 'around', 'chest', 'flowy', 'around', 'lower', 'half', 'would', 'say', 'running', 'big', 'strap', 'pretty', 'could', 'easily', 'nightwear', '5', '6', 'came', 'knee']	314
['nice', 'choice', 'holiday', 'gathering', 'like', 'length', 'graz', 'knee', 'conservative', 'enough', 'office', 'related', 'gathering', 'size', 'small', 'fit', 'well', 'usually', 'size', 'small', 'bust', 'opinion', 'run', 'small', 'larger', 'bust', 'definitely', 'size', 'perhaps', 'waist', 'big', 'problem', 'dress', 'quality', 'fabric', 'terrible', 'delicate', 'netting', 'type', 'fabric', 'top', 'layer', 'skirt', 'got', 'stuck', 'zip']	440
['took', 'package', 'wanted', 'fit', 'badly', 'could', 'tell', 'put', 'would', 'figure', 'straight', 'waist', 'way', 'small', 'body', 'shape', 'even', 'sized', 'could', 'tell', 'would', 'still', 'tight', 'waist', 'roomy', 'hip', 'said', 'really', 'nice', 'sturdy', 'fabric', 'pretty', 'color', 'well', 'made', 'hope', 'make', 'someone', 'happy']	345
['material', 'color', 'nice', 'leg', 'opening', 'large', '5', '1', '100', 'length', 'hit', 'right', 'ankle', 'leg', 'opening', 'size', 'waist', 'hem', 'line', 'ankle', 'front', 'pleat', 'make', 'fluffy', 'think', 'imagine', 'flattering', 'look', 'least', 'average', 'height', 'taller', 'may', 'look', 'good']	308
['took', 'chance', 'blouse', 'glad', 'crazy', 'blouse', 'photographed', 'model', 'paired', 'whit', 'white', 'pant', 'worked', 'perfectly', 'crisp', 'clean', 'would', 'describe', 'launders', 'well', 'fit', 'great', 'drape', 'perfect', 'wear', 'tucked', 'ca', 'go', 'wrong']	272
['flattering', 'super', 'cozy', 'coat', 'work', 'well', 'cold', 'dry', 'day', 'look', 'good', 'jean', 'dressier', 'outfit', '5', '5', '135', 'small', 'fit', 'great']	165
['love', 'look', 'feel', 'tulle', 'dress', 'looking', 'something', 'different', 'top', 'new', 'year', 'eve', 'small', 'chested', 'top', 'dress', 'form', 'fitting', 'flattering', 'look', 'steamed', 'tulle', 'perfect', 'ordered', 'xsp', 'length', 'perfect']	255

Table 14 shows that the decrease in word count following preprocessing implies that a number of text transformation and cleaning procedures were carried out to simplify the text data. Preprocessing frequently entails getting rid of stop words, which are regular words like "the," "and," and "is" that frequently don't add much to the text's meaning. Eliminating stop words may result in fewer words overall. Word counts may be lowered by custom steps like eliminating words, fixing spelling mistakes, or attending to domain-specific requirements, depending on the precise preprocessing techniques used.

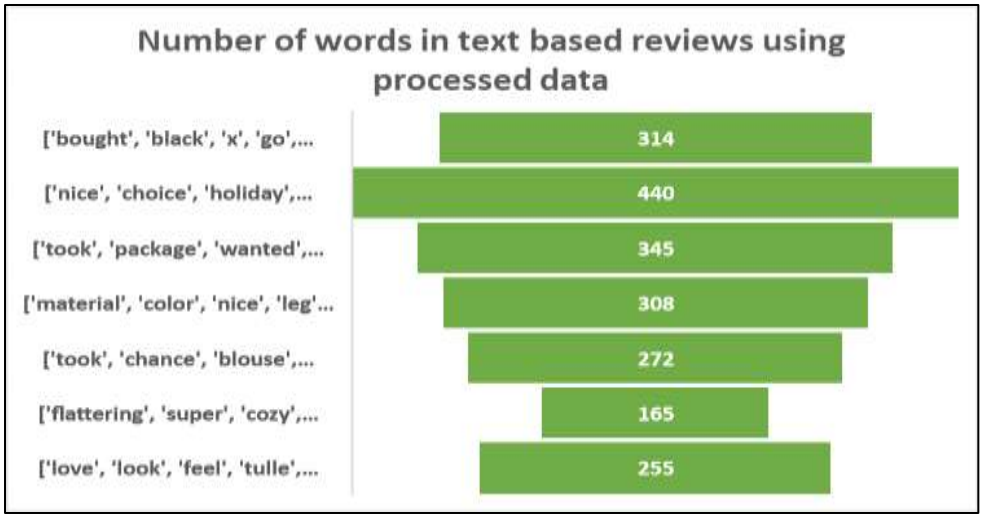

Figure 10: Number of words present in the Reviews after preprocessing

Smaller text sizes make analysis simpler and more efficient. Without being overtaken by superfluous or redundant data, researchers and analysts can concentrate on the essential information. Improved model generalization may result from a more condensed representation of the text following preprocessing.

Table 15: Comparison of Number of words present in the Reviews before and after preprocessing

Number of words in reviews of Raw Data	Number of words in reviews of Processed Data
377	314
500	440
470	345
359	308
296	272
167	165
292	255

When it comes to words or patterns that might not be pertinent to the overall meaning or sentiment, the model is less likely to overfit. Smaller text sizes make analysis simpler and more efficient. Without being overtaken by superfluous or redundant data, researchers and analysts can concentrate on the essential information.

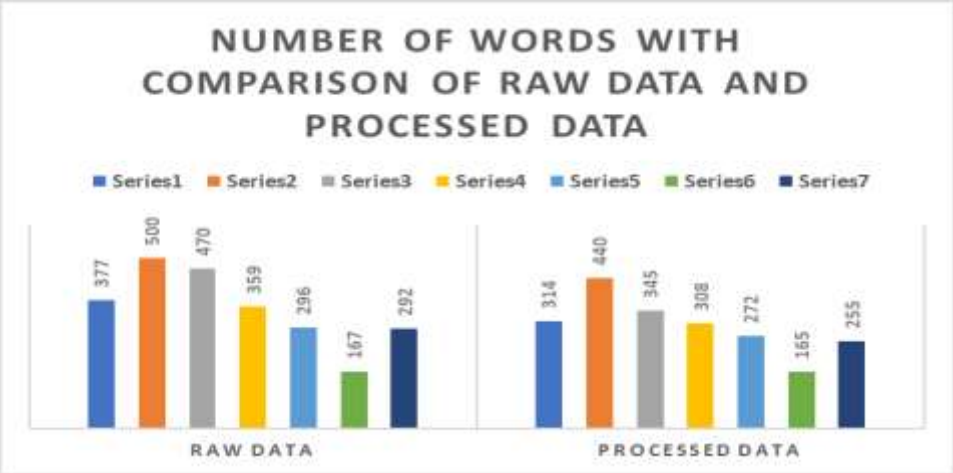

Figure 11: Comparison of Number of words present in the Reviews before and after preprocessing

Figure 11 shows by removing noise, normalizing the text, and extracting important features, preprocessing techniques help reduce the size of text data. This simplified representation improves the quality and efficacy of subsequent analytical tasks, like text classification or sentiment analysis, in addition to increasing the efficiency of processing and storing data.

7. Conclusion and Future Work

Based on the analysis of the electronic product dataset using both processed and unprocessed data, conclude that the four classification methods - Naive Bayes, Support Vector Machine (SVM), Random Forest (RF), and Decision Tree (DT) were applied. In both cases, Proposed ensemble method outperformed the other methods with the highest accuracy of 87% for the unprocessed data and 89% for the processed data. This indicates that RF is the best algorithm to use for this particular dataset. Naive Bayes and SVM showed the same accuracy, with 86% for both the unprocessed and processed data. Therefore, it can be concluded that processing the data has a positive impact on the accuracy of the algorithms used. It is clear from the comparison of different sentiment analysis algorithms on the provided dataset that the suggested ensemble approach—which combines Random Forest and XGBoost—performs better than Support Vector Machine (SVM) and Naive Bayes (NB). The accuracy findings show how well the ensemble approach captures intricate patterns in the data, improving sentiment classification. The suggested ensemble approach outperformed the competition with an accuracy of 89% by utilizing the advantages of Random Forest and XGBoost. By combining the predictive power of both models, the ensemble method yields a sentiment analysis solution that is more precise and all-encompassing.

References

[1] Patel, Aksh, Parita Oza, and Smita Agrawal, "Sentiment Analysis of Customer Feedback and Reviews for Airline Services using Language Representation Model", Procedia Computer Science 218, pp. 2459-2467, 2023

[2] Punetha, Neha, and Goonjan Jain, "Bayesian game model based unsupervised sentiment analysis of product reviews." Expert Systems with Applications 214, pp. 119128, 2023

[3] Joung, Junegak, and Harrison Kim, "Interpretable machine learning-based approach for customer segmentation for new product development from online product reviews", International Journal of Information Management 70 pp. 102641,2023.

[4] Rao, N. Srinivasa, and Chikkam Swathi. "Weakly-supervised deep learning for customer review sentiment classification." Journal of Engineering Sciences 14, no. 08, 2023.

[5] Diekson, Ziedhan Alifio, Muhammad Rivyan Bagas Prakoso, Muhammad Savio Qalby Putra, Muhammad Shaden Al Fadel Syaputra, Said Achmad, and Rhio Sutoyo. "Sentiment analysis for customer review: Case study of Traveloka." Procedia Computer Science 216, pp. 682-690, 2023.

[6] Solairaj, A., G. Sugitha, and G. Kavitha. "Enhanced Elman spike neural network based sentiment analysis of online product recommendation." Applied Soft Computing 132, pp. 109789, 2023.

[7] Rahman, Hameedur, Junaid Tariq, M. Ali Masood, Ahmad F. Subahi, Osamah Ibrahim Khalaf, and Youseef Alotaibi. "Multi-tier sentiment analysis of social media text using supervised machine learning." Comput. Mater. Contin 74,pp. 5527-5543, 2023.

[8] Omran, Thuraya M., Baraa T. Sharef, Crina Grosan, and Yongmin Li. "Transfer learning and sentiment analysis of Bahraini dialects sequential text data using multilingual deep learning approach." Data & Knowledge Engineering 143, pp. 102106, 2023.

[9] Chang, Victor, Lian Liu, Qianwen Xu, Taiyu Li, and Ching-Hsien Hsu. "An improved model for sentiment analysis on luxury hotel review." Expert Systems 40, no. 2, e12580, 2023.

[10] Singla, Zeenia, Sukhchandan Randhawa, and Sushma Jain, "Sentiment analysis of customer product reviews using machine learning", *International conference on intelligent computing and control (I2C2)*, IEEE, pp. 1-5, 2017.

[11] Kumar, KL Santhosh, Jayanti Desai, and Jharna Majumdar, "Opinion mining and sentiment analysis on online customer review", *International Conference on Computational Intelligence and Computing Research (ICCIC)*, IEEE, pp. 1-4, 2016.

[12] Jagdale, Rajkumar S., Vishal S. Shirsat, and Sachin N. Deshmukh, "Sentiment analysis on product reviews using machine learning techniques", *Cognitive Informatics and Soft Computing*, pp. 639-647, 2019.

[13] Kumari, Upma, Arvind K. Sharma, and Dinesh Soni, "Sentiment analysis of smart phone product review using SVM classification technique", *International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*, pp. 1469-1474. IEEE, 2017.

[14] Chauhan, Chhaya, and Smriti Sehgal, "Sentiment analysis on product reviews." *International Conference on Computing, Communication and Automation (ICCCA)*, IEEE, pp. 26-31, 2017.

[15] Janhavi, N. L., Jharna Majumdar, and Santhosh Kumar, "Sentiment Analysis of Customer Reviews on Laptop Products for Flipkart", *International Research Journal of Engineering and Technology (IRJET) 5*, no. 03, pp. 629-634, 2018.

[16] Pujari, Chetana, and Nisha P. Shetty. "Comparison of classification techniques for feature-oriented sentiment analysis of product review data", *Data Engineering and Intelligent Computing*, Springer, pp. 149-158, 2018.

[17] Shivaprasad, T. K., and Jyothi Shetty, "Sentiment analysis of product reviews: a review", *International Conference on Inventive Communication and Computational Technologies (ICICCT)*, pp. 298-301, 2017.

[18] Mukherjee, Subhabrata, and Pushpak Bhattacharyya, "Feature specific sentiment analysis for product reviews", *International Conference on Intelligent Text Processing and Computational Linguistics*, Springer, pp. 475-487, 2012.

[19] Fang, Xing, and Justin Zhan, "Sentiment analysis using product review data", *Journal of Big Data 2*, Vol. 1, pp. 1-14, 2015.

[20] Onan, Aytuğ, "Sentiment analysis on product reviews based on weighted word embeddings and deep neural networks, " *Concurrency and Computation: Practice and Experience 33*, Vol. 23, pp. e5909, 2021.

[21] Nguyen, Heidi, Aravind Veluchamy, Mamadou Diop, and Rashed Iqbal, "Comparative study of sentiment analysis with product reviews using machine learning and lexicon-based approaches", *SMU Data Science Review 1*, Vol. 4, pp. 7, 2018.