Sentiment Analysis Of Customer Reviews For Product Rating

1st Mugdha Malpe
Computer Engineering
Pune Institute Of Computer Technology
Pune, India
mdmalpe@gmail.com

2nd Parag Kulkarni
Computer Engineering
Pune Institute Of Computer Technology
Pune, India
parag.kulkarni.pk@gmail.com

3rd Amay Mapari
Computer Engineering
Pune Institute Of Computer Technology
Pune, India
amaymapari@gmail.com

4th Pratham Kolhe
Computer Engineering
Pune Institute Of Computer Technology
Pune, India
pmkolhe.pict@gmail.com

5th Abhi Deshmukh
Professor
Pune Institute Of Computer Technology
Pune, India
abhi.deshmukh@pict.edu

***Abstract*—Digital reviews play a crucial role in shaping consumer behavior and global communication. This research paper outlines a systematic approach to enhance product ratings using sentiment analysis. Utilizing classification models like Naïve Bayes, Support Vector Machine (SVM), and K-Nearest Neighbour, the study categorizes customer reviews into predefined positive or negative sentiment categories. The resulting ratings, on a scale of 1 to 5, provide prospective customers with a clear understanding of product reviews, aiding their decision-making process. This approach bridges the gap between textual data and meaningful product evaluations in the ever-growing digital marketplace.**

***Index Terms*—Sentiment Analysis,Support Vector Machine,Naïve Bayes, K-Nearest Neighbour,Vector Representation,Polarity Detection,Product Rating.**

I. Introduction

In the current times a huge surge is seen in the popularity of online shopping through E-commerce websites, where users can find everything they desire. Quality assurance has now shifted from traditional methods to user reviews and recommendations. These platforms provide an extensive space for users to share their experiences and opinions about products down to the smallest detail with others or potential customers.

This research study uses natural language processing (NLP) libraries like SpaCy and NLTK to conduct a thorough analysis of customer evaluations. These libraries help us classify consumer sentiment by making it easier to collect and process text data.In this paper, we investigate the efficacy of three different machine learning models in identifying consumer sentiment in product reviews: k-Nearest Neighbours (KNN), Naive Bayes, and Support Vector Machine (SVM). We intend to evaluate these models in terms of recall, accuracy, and precision when it comes to classifying evaluations into sentiment categories that are positive, negative, or neutral.The product is then given a rating out of five-stars based on these opinions.

II. Literature Review

In recent years, the analysis of data has emerged as a transformative tool, enabling humans to uncover hidden patterns within vast datasets. This ability to decipher and extract valuable insights from Big Data has become integral to the evolution of businesses, allowing them to better cater to the needs of their users. One promising source of structured and unstructured data for this purpose is online product reviews, where diverse consumers share their experiences with products. This review explores the potential of Big Data, defined by its five dimensions: volume, velocity, variety, veracity, and value. Veracity underscores the importance of data quality, while value centers on retaining the most pertinent information.[1]

To extract the valuable insights embedded in customer reviews, sentiment analysis, a technique within natural language processing, plays a pivotal role. Sentiment analysis focuses on discerning emotional content and opinions from textual data. This analysis can operate at multiple levels, including document, sentence, or even phrase levels. Two primary approaches are employed: subjective lexicon and machine learning. Various algorithms, such as SentiWordNet, Naive Bayes, and logistic regression, are leveraged to determine the emotions conveyed in online customer reviews. Evaluation metrics like precision, recall, and F-measure are applied to compare these algorithms, with Naive Bayes emerging as particularly effective.[2]

Furthermore, the study extends its investigation by comparing Naive Bayes and support vector machine (SVM) algorithms, utilizing a confusion matrix to evaluate their performance in predicting sentiment. The results highlight SVM as one of the superior algorithms for this task.Effective pre-processing of data is crucial for algorithmic efficiency, involving techniques like tokenization, stop word removal, and stemming to eliminate unnecessary language and misspellings.

To facilitate algorithmic readability, textual data must be transformed into numerical vectors. This conversion can be achieved through methods such as count vectorization and term frequency-inverse document frequency (TF-IDF).[3]

Moreover, the study underscores the utility of product reviews in identifying specific product features that resonate with customers, leading to the development of a sentiment classifier that captures feature-based emotions. This classifier helps to pinpoint critical product attributes, forming an assessment profile valuable to end-users.[4]

III. Dataset Description

This dataset consists of a few million Amazon customer reviews (input text) and star ratings (output labels) for learning how to train fastText for sentiment analysis.

The idea here is a dataset is more than a toy - real business data on a reasonable scale - but can be trained in minutes on a modest laptop.

1. Hyperparameters: - lr (Learning Rate): Set to 0.01, this hyperparameter determines the step size at each iteration during the model training process. It influences how quickly or slowly the model adapts to the data.

- dim (Dimension): This parameter, set to 20, represents the size of the word vectors (embeddings) used by the model. A lower dimension may lead to faster training but may capture less complex relationships in the data.

- epoch: The number of training epochs, set to 15. An epoch is one complete pass through the entire training dataset. More epochs can allow the model to learn from the data more thoroughly but may risk overfitting.

- wordNgrams: Set to 2, this parameter defines the maximum length of word n-grams used by the model. Word n-grams capture contextual information and relationships between words within a specified window.

- verbose: A binary parameter set to 1, indicating that training progress updates will be printed during the training process.

2. Model Training: - The FastText library is used to train a supervised classification model. - The training data is loaded from a CSV file. - The hyperparameters specified in the hyperparams dictionary are passed to the trainsupervised function using the unpacking operator (**). - The trained model is stored in the variable named model.

IV. Methodology

Data Preprocessing

A. Data Cleaning

This step involves eliminating duplicates, digits, punctuation marks, and numbers from the data. Stop-words and non-essential words are removed to improve accuracy.

B. Transformation and Tokenization

Data is divided into smaller units called tokens.

C. Removing Stop-Words

Common words with low semantic value are removed to prevent them from influencing the analysis.

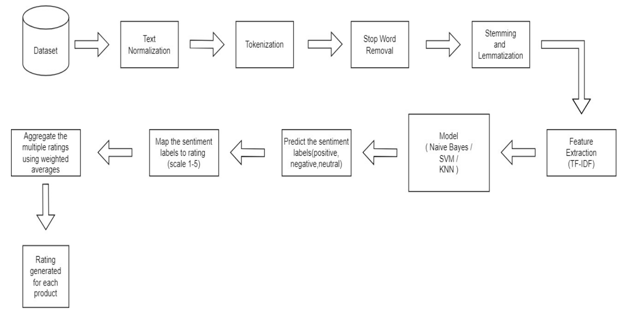

Fig. 1. System Architecture

D. Stemming

Words are reduced to their root forms by removing prefixes, suffixes, and infixes.

E. Term Weighting

After transforming words into terms, a method for representing documents based on these terms is determined. The term frequency–inverse document frequency (TF-IDF) feature weighting algorithm is commonly used for this purpose, enabling documents to be represented as vectors.

F. Pruning of Words

To simplify the dataset, less frequent features are filtered out, as words appearing less than two times in the documents are removed. This helps reduce the dimensionality of the term vector.

G. Feature Selection

Feature selection is critical for managing the high dimensionality of the feature space. It involves selecting a subset of features based on their importance. This can improve the efficiency and effectiveness of sentiment classification.

H. Principal Component Analysis (PCA)

PCA is a technique for dimensionality reduction while retaining most of the data's variance. It involves converting data into numerical form, calculating the covariance matrix, finding Eigen values and vectors, and selecting the top vectors to train and test the reduced datasets.

Sentiment Analysis Models

I. BERT

Google unveiled BERT, or Bidirectional Encoder Representations from Transformers, a ground-breaking paradigm for natural language processing in 2018. BERT, which is based on the Transformer architecture, stands out for its bidirectional training, which takes into account the context of both sides of a word during pre-training on massive text corpora. Because of this, the model can comprehend subtleties in context and record complex linguistic dependencies. BERT's performance can be enhanced for a variety of NLP tasks by utilizing its pre-learned representations, which it can use to refine for pre-training. Examples of these tasks include text categorization and question answering. The model's popularity has had a big impact on NLP, spurring the creation of more transformer-based models in the future.

J. spaCy

This study investigates how to improve the accuracy of product ratings by sentiment analysis of customer reviews using SpaCy, a powerful natural language processing (NLP) toolkit. SpaCy's sophisticated functionalities, such as dependency parsing, named entity recognition, tokenization, and part-of-speech tagging, provide a sophisticated comprehension of language structures and context, enhancing the accuracy of sentiment analysis. Through tokenization, grammatical context consideration, named entity identification, and the discovery of syntactic links, SpaCy aids in a more comprehensive sentiment analysis. It promises more accurate product ratings and helps businesses make decisions based on consumer feedback thanks to its effectiveness in processing massive datasets, contextual comprehension, and language support. As such, it is a beneficial addition to the sentiment analysis pipeline.

K. Support Vector Machines

Support Vector Machine(SVM) is a supervised learning method used for classification tasks. It is a valuable approach for finding the most effective boundary that separates positive and negative samples. The fundamental objective of SVM during training is to identify a hyperplane with the maximum margin to solve the classification task related to feature reviews. When distinguishing between two classes, there are countless potential boundaries to choose from. To make the best choice, it's crucial to opt for a decision boundary that maximizes the distance between any data points from both classes. Such a decision boundary with a maximum margin is less likely to lead to prediction errors, especially near the boundaries of either class, as highlighted by Ali, Kwak, and Kim (2016). In this context, the dot kernel type was preferred for its superior performance compared to radial and polynomial kernel types.

L. Naive Bayes (NB)

The Naive Bayes Tree (NB Tree) is a supervised classification algorithm that combines the principles of Bayesian probability with decision trees. This method utilizes Bayes' rule to calculate the likelihood of each class for a given set of instances. It does so under the assumption that the attributes are conditionally independent of the label.In simpler terms, a Naive Bayes classifier makes a "naive" assumption that the presence or absence of a particular feature (attribute) in a class is unrelated to the presence or absence of any other feature. For instance, in classifying a fruit, it would consider features like color, roundness, and size independently, even if they are related. This classifier calculates the probability of an object belonging to a class based on these independent contributions.

The advantage of the Naive Bayes classifier lies in its efficiency and ability to work with relatively small amounts of training data. It estimates the means and variances of the variables needed for classification, and due to the assumption of independence, it only requires determination of the variances of the variables for each label, rather than dealing with the entire covariance matrix.

M. K-Nearest Neighbors (KNN)

The KNN algorithm is a widely recognized instance-based approach that is frequently employed in text categorization due to its simplicity and effectiveness. When tasked with categorizing an unknown document, the KNN classifier examines the document's neighbors from the training set and relies on the class labels of the k most similar neighbors for its decision. The similarity between two documents can be computed using various metrics like Euclidean distance, cosine similarity, and more. The similarity score of each nearest neighbor document to the test document is employed as the weight for the classes of the neighbor document. If a specific category is shared by more than one of the k-nearest neighbors, the weight of that shared category is determined by summing the similarity scores of those neighbors, as detailed by Uğuz (2011).In the classification phase using KNN, the key parameter influencing classification is the number of nearest neighbors, often denoted as 'k.' Typically, the optimal value for k is determined empirically. In our study, the value of k was selected to optimize classification accuracy (k = 1 was chosen). Additionally, during the process of identifying the k-nearest neighbors, a mixed Euclidean distance metric is employed for distance measurement.

Feature Engineering

N. TF-IDF Vectorizer

In order to improve the precision and comprehensibility of sentiment evaluations, this study integrates the Term Frequency-Inverse Document Frequency (TF-IDF) vectorizer into the sentiment analysis framework for customer reviews. A key method in natural language processing, TF-IDF gives words weights according to how frequently they appear in a document and throughout the corpus, giving textual data a more complex representation. In order to guarantee that the sentiment analysis model concentrates on discriminative and contextually significant phrases, TF-IDF prioritises words that are both common inside a document and uncommon over the whole dataset. This helps the model capture the nuances present in customer evaluations.The integration of TF-IDF into the sentiment analysis pipeline contributes to a more sophisticated understanding of sentiment expressions, positioning it as a valuable tool for improving the precision of sentiment analysis in the realm of customer feedback.

O. CountVectorizer

The CountVectorizer is a crucial feature extraction technique in the natural language processing (NLP) domain of machine learning when it comes to sentiment analysis for customer evaluations. The text data is converted into a bag-of-words representation via CountVectorizer, where each document is represented by a vector of word counts. It calculates the frequency of every word in a document and gives machine learning models a numerical representation that makes sense. Word occurrence in each document is captured by CountVectorizer, which turns text into a high-dimensional matrix. This enables the sentiment analysis model to identify patterns and connections between words and sentiments.This technique is particularly valuable for its simplicity and efficiency, making it well-suited for tasks such as sentiment analysis in customer reviews where the emphasis is on word frequency as a primary indicator of sentiment.

V. Model Evaluation and Results

Within the project, evaluating the models entails determining how well the three classifiers—Support Vector Machine (SVM), k-Nearest Neighbours (K-NN), and Naive Bayes—perform in terms of prediction accuracy. The purpose of this examination is to identify the best classifier for categorizing customer feedback. The models will be evaluated using common measures like accuracy, precision, recall, and F1-score to see how well they identify reviews as positive or negative. The model most suited for effectively assessing sentiment in customer reviews will be the one with the highest accuracy and the best precision/recall ratio.

A. Score Computation

1) Assign Sentiment Labels to Numerical Values: Establish a systematic mapping between sentiment labels and numerical values that align with the chosen rating scale. This mapping allows sentiment labels to be quantified, thereby enabling their inclusion in rating calculations.

2) Calculate Individual Review Ratings: For each customer review, apply the predefined mapping to determine the corresponding numerical rating value. This mapping serves as the bridge between textual sentiment and numerical representation.

3) Aggregate Ratings: When dealing with multiple reviews for a single product, the aggregate rating must be determined. This can be achieved through various methods, such as averaging, weighted averaging, using medians, or creating custom aggregation rules. The chosen approach should align with the specific project requirements and the nature of the data.

4) Round the Aggregate Rating: The final step involves rounding the aggregate rating to ensure it conforms to the specified rating scale, resulting in a whole number within the prescribed range.

This method enables the transformation of sentiment labels extracted from customer reviews into meaningful product ratings, facilitating the communication of the overall sentiment and user experience associated with a product. It is essential to emphasize that, in practical applications, a nuanced approach considering various factors may be employed to calculate the most accurate and informative product ratings.

This final score would be in a range of 0-5 (0 being the worst score - 5 being the best score). The customer then can make a calculated decision of whether he wants to buy the product or not.

B. Attribute-based Analysis

Along with the overall score for the product, providing the user with attribute-based analysis helps the user in determining whether his need for a certain attribute is met in the product in consideration or not. Determining the occurrence of specific words that correspond to a feature or attribute will help the user to understand the key aspects of the product. This helps the user to understand the relationship between the different aspects of the product and the reviews associated with it. This is implemented by maintaining a dictionary containing all the important features of any product and then checking the reviews for these features. If any of these features are mentioned in the reviews, its computed sentiment score is taken into consideration and then an overall score for that specific feature is computed, thus providing a score for each of the features of a product.

VI. Challenges

Sentiment analysis in the context of product ratings presents several challenges that need to be addressed to ensure the accuracy and effectiveness of the analysis. In this section, we discuss some of the major challenges associated with sentiment analysis in the context of customer reviews for product ratings.

Data Quantity and Quality: Obtaining high-quality labeled data for training is difficult, and ensuring data represents diverse products and user sentiments is challenging.

Multi-Lingual Analysis: Handling international user bases with diverse languages and cultural nuances is a significant challenge.

Context and Sarcasm: Identifying sentiments in cases of irony, sarcasm, and context-dependent attitudes is complex due to subtleties.

Privacy and Ethical Concerns: Balancing sentiment analysis with user privacy and ethical issues, such as prediction bias, is a critical challenge.

These challenges must be carefully considered and addressed in the development of sentiment analysis models for product ratings to improve the reliability and applicability of the results.

VII. Conclusion

Our study offers insightful information about the field of sentiment analysis for product ratings. We've looked at sentiment analysis's approaches, strategies, and uses, showing how important it is for comprehending consumer views and influencing corporate choices. Our study adds to the field's continuous development by illuminating the shortcomings and efficacy of current models. Our research is a first step toward enhancing the precision and usefulness of sentiment analysis for product ratings, as companies and consumers depend more and more on it to make informed decisions. In the end, we hope that our findings spur more investigation and creativity in this field, improving user experiences and supporting more data-driven business decision-making.

VIII. Future Scope

Sentiment analysis models have a bright future ahead of them as their capabilities and uses grow. These models will improve their multilingual and cross-cultural competence, taking into account a variety of languages and cultural quirks. They will also explore more complex emotional nuanced aspects than only positive and negative emotions, such as sarcasm and irony. In an age of plentiful information and social media contacts, they will excel at comprehending context within talks and enabling real-time analysis. More insightful sentiment analysis algorithms tailored to certain businesses will continue to emphasize ethical issues like privacy and bias identification. These models will be essential to improving user experiences across a range of applications, facilitating easy integration with other AI technologies, and driving advancements in the field of natural language processing.

References

[1] Pang, B., Lee, L. (2008). Opinion mining and sentiment analysis. Foundations and Trends® in Information Retrieval, 2(1-2), 1-135

[2] Liu, B. (2012). Sentiment analysis and opinion mining. Synthesis Lectures on Human Language Technologies, 5(1), 1-167.

[3] Hu, M., Liu, B. (2004). Mining and summarizing customer reviews. In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 168-177).

[4] Popescu, A. M., Etzioni, O. (2007). Extracting product features and opinions from reviews. In Natural Language Processing and Text Mining (pp. 9-28). Springer.

[5] Liu, K., Xu, Y. (2015). Mining opinions from critical product reviews. Data Knowledge Engineering, 98, 29-43.

[6] Zhang, L., Zhang, F., Yao, J. (2015). Sentiment analysis for Chinese microblog with semantic expansion. Knowledge-Based Systems, 69, 63-76.

[7] Qu, L., Ifrim, G., Weikum, G. (2010). The bag-of-opinions method for review rating prediction from sparse text patterns. In Proceedings of the 23rd international conference on computational linguistics (pp. 913-921).