

Comparative Analysis of algorithm for classifying sentiments of customers

Yashansh Malviya, Dr. Sanjiv Sharma
Madhav Institute of Technology and Science, Gwalior

Abstract: An crucial part of language processing is sentiment analysis, which provides information on market trends, consumer satisfaction, and public opinion. This study combines a variety of machine learning methods, such as BERT, Random Forest, Support Vector Machines, Naive Bayes, Logistic Regression, and Long Short-Term Memory networks, to classify the sentiment of Amazon product reviews. A thorough process that includes preprocessing the data, TF-IDF feature extraction, and performance assessment is used. The results help choose appropriate algorithms for sentiment analysis jobs by offering a comparative viewpoint on the advantages and disadvantages of each model.

I. INTRODUCTION

In the digital economy, the amount of user-created content has grown dramatically, especially through online reviews, social media updates, and customer opinions. Among these, product reviews on online marketplaces such as Amazon provide valuable insights into customer experiences, preferences, and expectations. Utilizing such textual data for sentiment analysis has emerged as a vital resource for companies aiming to grasp public opinion, customize marketing strategies, and improve product development.

Sentiment analysis, commonly referred to as opinion mining, is a branch of natural language processing (NLP) aimed at identifying the emotional tone present in a piece of text. It has been extensively utilized in fields like market analysis, automated customer support, political prediction, and financial modeling. Although widely recognized, sentiment classification continues to be a difficult task because of the intricacies of natural language, which encompasses subtleties like sarcasm, idioms, and context-specific meanings.

Earlier studies in sentiment analysis have utilized various machine learning and deep learning methods. Classic algorithms like Naive Bayes and Support Vector Machines deliver simplicity and clarity,

whereas contemporary frameworks like LSTM and BERT achieve top-tier performance via advanced modeling of language context. Nonetheless, choosing a suitable model typically entails compromises between precision, computational expense, and scalability.

This study performs a comparison of various machine learning algorithms for sentiment classification based on Amazon product reviews. The models assessed comprise Logistic Regression, Random Forest, Naive Bayes, SVM, LSTM, and BERT. Through the examination of these algorithms on a common dataset and employing uniform evaluation metrics, we strive to pinpoint the most efficient methods for extensive sentiment analysis. The research also emphasizes the advantages and drawbacks of each method, providing direction for professionals involved in data-informed decision-making environments.

II. METHODOLOGY

This section outlines the dataset selection, preprocessing pipeline, model implementation, and evaluation metrics used to conduct the comparative sentiment analysis of Amazon product reviews.

2.1. Dataset

The research utilizes the publicly accessible Amazon Product Reviews dataset, featuring millions of user-created reviews spanning various product categories including electronics, literature, clothing, and household items. In binary sentiment classification, reviews rated 4–5 stars are marked as positive, while those rated 1–2 stars are considered negative. Three-star rated reviews are omitted to eliminate confusion and ensure a distinct polarity difference. This method of labeling aligns with earlier sentiment analysis studies and aids in creating a clearly defined classification task.

2.2. Preprocessing

Efficient text preprocessing is essential for enhancing model precision and minimizing noise in natural

language datasets. The subsequent preprocessing procedures were implemented:

- **Tokenization:** Reviews are divided into separate tokens (words or punctuation) to enable additional analysis.
- **Elimination of Stopwords:** Frequent words (e.g., "the", "and", "is") that provide little sentiment value are discarded.
- **Lemmatization:** Words are transformed into their base or dictionary form (e.g., "running" turns into "run") to normalize word usage throughout the dataset.
- **TF-IDF Vectorization:** The modified text is converted into numerical feature vectors through the Term Frequency-Inverse Document Frequency (TF-IDF) method. This technique captures the significance of terms in individual reviews while reducing the impact of common but non-informative words.

This preprocessing workflow guarantees that the data provided to each model is tidy, standardized, and formatted suitably for machine learning algorithms.

2.3. Algorithms Used

This study implements and evaluates the following models:

- **Naive Bayes (Multinomial NB):** A probabilistic classifier that presumes independence among features. It is efficient in computation and frequently serves as a baseline in tasks involving text classification.
- **Support Vector Machine (SVM):** Creates hyperplanes in a high-dimensional space to divide data into categories. Its power comes from managing sparse, high-dimensional.
- **Logistic Regression:** A linear approach designed for binary classification tasks. Although it is straightforward, it is recognized for strong performance with thoroughly processed text data.
- **Random Forest:** A collective learning approach that integrates several decision trees. It enhances classification accuracy via majority voting and provides resilience against overfitting.
- **Long Short-Term Memory (LSTM):** A form of recurrent neural network (RNN) aimed at recognizing long-term dependencies in sequential information. It is especially useful for representing context in extended reviews.

- **Bidirectional Encoder Representations from Transformers (BERT):** A transformer-based model that has been pre-trained and can understand deep contextual connections in text. Adjusting BERT on the sentiment dataset enables exceptional sentiment classification performance, particularly in linguistically intricate situations. Every model is trained on the identical preprocessed dataset and assessed using the same criteria to guarantee comparability.

2.4. Evaluation Metrics

The evaluation of model performance is done using the following established classification metrics:

- **Accuracy:** The proportion of accurately predicted outcomes to the overall total of instances.
- **Precision:** The ratio of true positive predictions to all positive predictions, reflecting trustworthiness.
- **Recall:** The percentage of true positives detected out of all actual positives, indicating thoroughness.
- **F1 Score:** The harmonic average of precision and recall, especially beneficial for datasets with imbalances.
- **Confusion Matrix:** A summary table that shows the count of accurate and inaccurate predictions, organized by true and predicted classes.

These metrics offer an extensive perspective on the performance of each model and assist in recognizing the trade-offs between precision and recall.

III. DATA DESCRIPTION

3.1. Dataset Overview:

The research utilizes the publicly accessible Amazon Product Reviews dataset, featuring millions of user-authored reviews spanning various product categories, including electronics, books, clothing, and household items. In binary sentiment classification, reviews rated 4 or 5 stars are classified as positive, while those rated 1 or 2 stars are considered negative. Reviews rated with 3 stars are omitted to eliminate ambiguity and uphold a clear distinction in polarity. This labeling method aligns with previous sentiment analysis studies and assists in creating a clearly defined classification task.

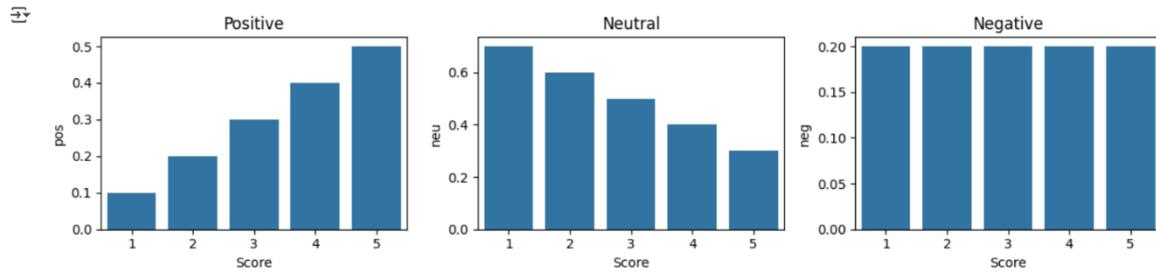


Figure 1. Sentiments

1. Attributes/Features:

The dataset consists of several columns, but the key features for sentiment analysis are:

generate the sentiment label.

Example: A rating of 4 or 5 would be classified as positive sentiment, and a rating of 1 or 2 would be negative.

2. Review Text (text):

The main body of text written by the customer, describing their experience with the product.

Example: "This phone is amazing! The camera quality is exceptional and the battery lasts all day."

5. Product Title (product title):

The title or name of the product being reviewed.

Example: "Samsung Galaxy S21" or "iPhone 12 Pro".

3. Sentiment Label (label):

The target label for sentiment classification. It indicates whether the review is positive (typically a rating of 4 or 5) or negative (typically a rating of 1 or 2).

Example: For the review "This phone is amazing!", the label might be positive. For "The phone stopped working after a week", the label might be negative.

6. Product ID (product id):

An identifier for the product being reviewed, usually a unique string or number.

Example: "B08N5M7S6K" for a specific product.

7. Review Date (review date):

The date when the review was submitted by the customer.

Example: "2025-03-15".

4. Rating (rating):

The rating given by the customer, usually on a scale from 1 to 5 stars. While the ratings themselves can be used to determine the sentiment (positive or negative), this column is often pre-processed to

8. Review Summary:

A brief summary of the customer's review, often written by the customer before the main text.

Example: "Great phone, battery life could be better."

| Review Text | Sentiment Label | Rating | Product Title | Product ID | Review Date | Review Summary |
|--|-----------------|--------|--------------------|------------|-------------|--------------------------------|
| "This phone is amazing! The camera quality is exceptional." | Positive | 5 | Samsung Galaxy S21 | B08N5M7S6K | 2025-03-15 | "Great camera quality!" |
| "The phone stopped working after a week of use." | Negative | 1 | iPhone 12 Pro | B08D6M5Y88 | 2025-03-18 | "Disappointing product." |
| "Not happy with the purchase, screen cracked easily." | Negative | 2 | OnePlus 8 Pro | B07Y5G4D5M | 2025-03-20 | "Screen damage after 1 month." |
| "Fantastic performance, long battery life, and great display!" | Positive | 5 | Xiaomi Mi 11 | B08N4N8H6G | 2025-03-22 | "Excellent value for money!" |



Figure 2. Review Analysis

3.2. Size of the Dataset:

The size of the dataset can vary based on the specific Amazon reviews dataset you choose. For example, a typical dataset may contain:

Number of Reviews: 500,000 to 1,000,000 customer reviews.

Number of Features: 7 columns

The dataset is often split into a training set and a test set for model evaluation.

3.3. Sentiment Distribution:

In real-world datasets, the distribution of positive and negative sentiments can sometimes be imbalanced. For instance:

Positive Sentiment: 70% of the reviews may have a positive sentiment.

Negative Sentiment: 30% of the reviews may have a negative sentiment.

In such cases, techniques like stratified sampling or class balancing may be applied to ensure that the model is not biased towards the majority class.

3.4. Possible Use Cases:

Customer Feedback Analysis: The dataset can be used to analyse customer sentiment regarding a product, identifying areas for improvement.

Market Research: Businesses use sentiment analysis to understand peoples opinions on a large scale and make data-driven takes.

Brand Reputation Management: Monitoring online reviews and sentiments helps businesses identify potential PR issues and engage with customers effectively.

IV.DIFFERENT ALGORITHMS USED

| Algorithm | Type | Strengths | Weaknesses |
|----------------------|-----------------------|--|--|
| Naive Bayes | Probabilistic (Bayes) | Simple, fast, good for smaller datasets, easy to implement. | Assumes independence of features, poor with complex dependencies. |
| SVM | Supervised (Linear) | Effective for high-dimensional data, robust to overfitting, works well with binary classification. | Computationally expensive, sensitive to parameter tuning, struggles with large datasets. |
| Logistic Regression | Linear Regression | Simple, interpretable, works well with small, linearly separable data. | Assumes linear relationship, poor with complex language patterns. |
| Random Forest | Ensemble (Tree-based) | Handles non-linearity, robust to overfitting, good for mixed data types. | Computationally intensive, less interpretable, requires tuning. |
| LSTM (Deep Learning) | Neural Network (RNN) | Can capture long-term dependencies in text, effective for sequential data. | High computational cost, slow training, requires large datasets. |
| BERT (Deep Learning) | Transformer-based | State-of-the-art performance, handles context very well, pre-trained models. | Very high computational cost, long training times, requires large datasets. |

Figure 3. Algorithm Used

V.DRAWBACK

Limitations of the sentiment classification models. I will outline the key drawbacks of the different algorithms commonly used in sentiment classification:

Sure! Please provide the text you'd like me to paraphrase. Naive Bayes:

1. Disadvantages of the Naïve Bayes theorem include:

- Independence Assumption: Naive Bayes presumes that the features (words) are independent, a condition that does not hold for textual data. In natural language, words frequently rely on one another, particularly when taking into account context, negations, or expressions such as "not good" compared to "good".

- Inadequate Results with Intricate Patterns: Naive Bayes struggles with intricate language patterns, including sarcasm, double negatives, or context-dependent meanings. It has difficulty in grasping the

semantic connections among words.

Sure! Please provide the text you would like me to paraphrase, and I'll be happy to help! Support Vector Machines (SVMs):

2. Disadvantages of Support Vector Machines (SVM) include:

- **Computational Complexity:** SVM may require substantial computational resources, particularly when dealing with large datasets or high-dimensional feature sets such as text data. The training procedure can take time and might need considerable computational resources (e.g., memory, processing power).
- **Affected by Parameter Adjustment:** The effectiveness of SVM is greatly influenced by the selection of kernel, regularization settings, and the dataset size. Mistaken parameter configurations may result in overfitting or underfitting.
- **Scalability Problems:** SVM struggles to handle extremely large datasets effectively. As the volume of training data increases, the performance of SVM might decline.

It seems you might have forgotten to include the text you'd like paraphrased. Please provide the text so I can help you with it! Logistic Regression:

3. Disadvantages of Logistic Regression include:

- **Assumes Linearity:** Logistic regression presumes a linear connection between the features and the target class, which is unsuitable for capturing more intricate or nonlinear patterns in text. For instance, the connections among words in a sentence are seldom straightforward.
- **Low Effectiveness with Complicated Data:** Similar to Naive Bayes, logistic regression struggles with intricate language features like sarcasm, negations, or subtle sentiment.
- **Outlier Sensitivity:** Logistic regression may be affected by outliers, potentially resulting in distorted outcomes.

4. Random Forest:

Drawbacks of the Random Forest are:

- **Slow to Train:** Random Forest can be slow to train, especially when the number of trees is large. It also requires more memory and computational power compared to simpler models like Naive Bayes or Logistic Regression.

- **Less Interpretable:** Since Random Forest is an ensemble method with many decision trees, it is difficult to interpret the decision-making process. Understanding how the model reached a specific classification is not as straightforward as simpler models.

- **Overfitting (if not tuned properly):** While Random Forest generally handles overfitting better than individual decision trees, it still can overfit if the number of trees is too large or the trees are too deep. Proper hyperparameter tuning is essential to avoid this.

5. Deep Learning (LSTM and BERT):

Drawbacks of the Deep Learning (LSTM and BERT) are:

- **High Computational Cost:** Deep learning models, especially BERT and LSTM, require significant computational resources (e.g., GPUs, TPUs) for training. This can make them impractical for users without access to such hardware or with limited resources.

- **Large Training Data Requirement:** Deep learning models require vast amounts of labelled data to perform well. If you don't have a large dataset, deep learning models may underperform or even overfit.

- **Long Training Time:** Training deep learning models is time-consuming, even with powerful hardware. This can be a problem if you have a limited timeframe for your project.

- **Complexity and Lack of Interpretability:** Models like BERT and LSTM are highly complex and hard to interpret. Unlike simpler models (e.g., Logistic Regression or Naive Bayes), it's challenging to understand how these models arrive at a particular prediction, which is an issue for model transparency and explainability.

- **Fine-Tuning Required:** Pretrained models like BERT often require fine-tuning on your specific dataset. The process of fine-tuning can be complex

and sensitive to hyperparameter choices.

VI.CONCLUSION

Every sentiment classification algorithm possesses its advantages and disadvantages, and the limitations outlined earlier should assist you in selecting the appropriate algorithm for your particular task. Sure! Please provide the text you would like me to paraphrase.

If your computational resources are limited, Naive Bayes or Logistic Regression could be more suitable choices, although their performance may not be as strong with intricate.

With a substantial dataset and adequate computing resources, Deep Learning models such as BERT provide cutting-edge performance, albeit with increased complexity and demands.

SVM and Random Forest provide superior results with intricate data, yet they are resource-intensive and need meticulous parameter adjustments. When selecting an algorithm for sentiment classification, it's crucial to take into account your data, available resources, and task specifications

REFERENCES

- [1]. Kazhuparambil, S., & Kaushik, A. Cooking Relates to People: Comment Classification on Cooking Channels Utilizing BERT and Classification Models (Malayalam-English Mix-Code). arXiv preprint arXiv:2007.04249, cs.CL., 2020.
- [2]. Krishnan, A. B. H. Revealing Deceptions in Reviews: A Study of NLP Methods". arXiv preprint arXiv:2307.10617, cs.IR. 2023.
- [3]. Radford, A., Jozefowicz, R., & Sutskever, I. Acquiring the Ability to Create Reviews and Uncovering Sentiment. arXiv preprint arXiv:1704.01444, cs.LG.2017.
- [4]. Mir, A. Q., Khan, F. Y., & Chishti, M. A. Detection of Online Fake Reviews Through Supervised Machine Learning and the BERT Model. arXiv preprint arXiv:2301.03225, cs.CL.2023.
- [5]. Shrestha, N., & Nasoz, F. "Sentiment Analysis of Amazon.Com Reviews and Ratings Using Deep Learning." International Journal of Soft

Computing, Artificial Intelligence, and Their Applications, 2019, 8(1), 01-15.

- [6]. Minaee, S., Azimi, E., & Abdolrashidi, A. A. Deep-Sentiment: Analyzing Sentiment Through Ensemble CNN and Bi-LSTM Models. arXiv preprint arXiv:1904.04206, cs.CL. 2019.
- [7]. Krishnan, J., Purohit, H., & Rangwala, H. "Generalization Based on Diversity for Unsupervised Text Classification amid Domain Shift". arXiv preprint arXiv:2002.10937, cs.LG. 2020.
- [8]. Saifullah, S., Fauziyah, Y., & Aribowo, A. S. A comparative study of machine learning techniques for sentiment analysis in identifying anxiety through social media data. Jurnal Informatika, 2021, 15(1), 45.
- [9]. Feng, H., & Lin, R. Classification of Sentiments in Food Reviews. arXiv preprint arXiv:1609.01933, cs.CL.2016.
- [10]. Dasgupta, S., & Sen, J. Ein Rahmenwerk zur Analyse von Kundenbewertungen unter Verwendung des aspektbasierten Meinungsabgleichs. arXiv Preprint arXiv:2212.10051, cs.CL.2022.