Formosa Journal of Computer and Information Science (FJCIS)
Vol. 5, No. 1 2026: 19-36

Comparative Study of Machine Learning Models for Sentiment Analysis of Amazon Product Reviews

Tri Noviantoro[1*], Suryaneta[2]
[1]Universitas Muhammadiyah Lampung
[2]Institut Teknologi Sumatera
Corresponding Author: Tri Noviantoro trinovi@uml.ac.id

A R T I C L E I N F O

Keywords: Amazon Product Reviews, BERT, Customer Behavior, Machine Learning, Sentiment Analysis

Received : 16, November
Revised : 18, January
Accepted: 20, March

A B S T R A C T

This research presents a comparative analysis of four popular sentiment classification models: Naive Bayes, Support Vector Machine (SVM), Long Short-Term Memory (LSTM) networks, and Bidirectional Encoder Representations from Transformers (BERT). The models are evaluated using the Amazon Product Reviews dataset based on their ability to classify sentiments into positive or negative categories. The results show that BERT outperforms the other models in accuracy, precision, recall, and F1-score, demonstrating its superior ability to capture complex contextual relationships in text. LSTM performed well, particularly in recalling positive sentiments, but was outperformed by BERT overall. Conversely, Naive Bayes and SVM exhibited lower accuracy and higher false positive rates, highlighting their limitations in handling nuanced, context-dependent text. This study emphasizes the trade-offs between traditional machine learning models and advanced deep learning techniques.

DOI prefik: https://doi.org/10.55927/fjcis.v5i1.16389
ISSN-E: 2830-3040
https://journal.formosapublisher.org/index.php/fjcis

INTRODUCTION

Businesses face the challenge of managing and interpreting millions of online product reviews published daily on platforms like Amazon in the era of e-commerce. Understanding the underlying sentiment of these reviews is crucial for decision-making and business management through big data study. Sentiment analysis, as one branch of the big data approach, is the process of determining the emotional tone behind a piece of text, and it has emerged as a powerful tool for understanding customer behavior. How to understand businesses can derive meaningful insights from customer feedback is crucial, as one of the methods can be achieved by leveraging natural language processing (NLP) techniques, which can inform product development, marketing strategies, and customer engagement (Liu et al., 2021; Mustak et al., 2024)

Recent advancements in NLP, particularly through deep learning models, have greatly improved the accuracy and efficiency of sentiment analysis tasks. Traditional machine learning models such as Naive Bayes and Support Vector Machines (SVM) have been widely used in text classification tasks, including sentiment analysis. These models are known for their simplicity and effectiveness when the text data is relatively structured (Sebastiani, 2002). However, as the complexity of text data increases, especially in product reviews that may contain ambiguous, nuanced, or contextual information, more advanced models are required (Yadav et al., 2024).

Long Short-Term Memory (LSTM) networks, specialized Recurrent Neural Network (RNN) type, have shown considerable promise in handling sequential data, making them particularly suitable for text classification tasks. LSTM networks can learn long-term dependencies in data, which is crucial for understanding context and sentiment in longer product reviews (Goldberg, 2016). On the other hand, Bidirectional Encoder Representations from Transformers (BERT), a transformer-based model, has set new benchmarks in the field of NLP. BERT's ability to capture context from both directions in a sentence, combined with its pre-training and fine-tuning capabilities, has revolutionized sentiment analysis tasks, making it one of the most effective models for various NLP applications, including sentiment analysis of product reviews (Devlin et al., 2018).

Despite the success of transformer models like BERT, there remains a need to compare the performance of traditional machine learning models with more advanced deep learning approaches. This paper aims to compare Naive Bayes, SVM, LSTM, and BERT in the context of sentiment analysis on Amazon product reviews. By examining these models' strengths and weaknesses, this research offers a comprehensive understanding of which techniques are most effective for sentiment analysis in the e-commerce domain.

This study will contribute to the growing body of literature on sentiment analysis by providing insights into the comparative performance of these models and exploring their broader implications for businesses and marketers. The findings could guide decision-making on which model to adopt based on the nature of the text data, available computational resources, and specific business objectives.

LITERATURE REVIEW

Sentiment analysis is a subfield of Natural Language Processing (NLP) that focuses on identifying, extracting, and classifying opinions or emotions expressed in textual data. In the context of e-commerce platforms such as Amazon, sentiment analysis plays a crucial role in understanding customer perceptions of products, which can support decision-making for both consumers and businesses. Early studies in sentiment analysis predominantly employed traditional machine learning algorithms such as Naïve Bayes, Support Vector Machine (SVM), and K-Nearest Neighbor (KNN). Pang and Lee (2008) highlighted that Naïve Bayes is particularly effective for text classification due to its simplicity and ability to handle high-dimensional data. Meanwhile, SVM has been widely recognized for its strong performance in binary classification tasks, as it identifies the optimal hyperplane for separating classes (Joachims, 1998).

With the advancement of computational techniques, deep learning approaches have gained significant attention in sentiment analysis. Models such as Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and Convolutional Neural Networks (CNN) have demonstrated superior capability in capturing contextual information within text sequences compared to traditional methods (Liu, 2012). In particular, LSTM is designed to address the vanishing gradient problem, enabling it to learn long-term dependencies in textual data more effectively.

More recently, transformer-based models such as BERT (Bidirectional Encoder Representations from Transformers) have significantly improved sentiment analysis performance. Devlin et al. (2019) showed that BERT captures bidirectional context in sentences, resulting in richer and more accurate text representations than previous models. In the case of Amazon product reviews, the availability of large-scale and diverse datasets presents both opportunities and challenges for model comparison. Several studies indicate that deep learning models generally outperform traditional machine learning approaches in terms of accuracy, although they require greater computational resources. On the other hand, traditional models remain relevant due to their efficiency and ease of implementation.

METHODOLOGY

This section outlines the research methods, including data collection, preprocessing, model development, and evaluation methods used to compare Naive Bayes, SVM, LSTM, and BERT for sentiment analysis on Amazon product reviews. The goal is to assess the effectiveness of traditional machine learning models versus deep learning approaches in capturing customer sentiment start by dataset collection, then preparation, followed by building and testing model and the last one performance comparison and summary as shown in Figure 1 below.

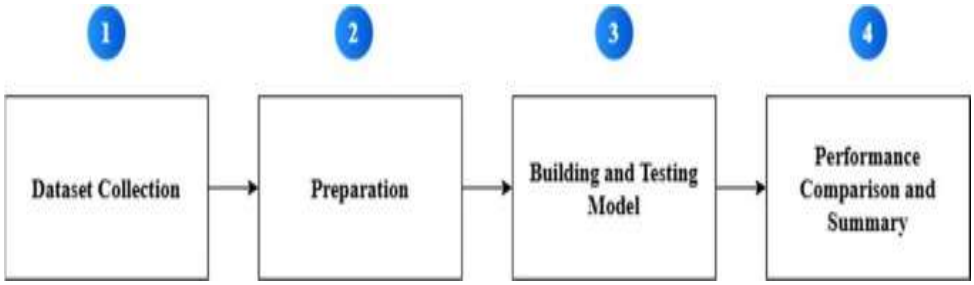

Figure. 1. Research method framework.

Natural Language Processing (NLP)

Over the past few decades, Natural Language Processing (NLP) has progressed considerably, transitioning from basic rule-based systems to advanced machine learning and deep learning methods. In sentiment analysis, the primary goal is determining the emotional tone behind a body of text. Classifying text as positive, negative, or neutral is crucial for understanding customer opinions and guiding decision-making in marketing, customer service, and product development.

In the past, early sentiment analysis models relied heavily on lexicon-based approaches, where predefined dictionaries of positive and negative words were used to classify text (Liu, 2020). Although these methods were simple and computationally efficient, they had difficulty handling context, ambiguity, and the nuances of natural language. As such, the NLP community shifted towards machine learning (ML) models, which could learn from labeled data to classify sentiments with higher accuracy.

In recent years, deep learning has led to significant advancements in NLP. Neural network models, particularly Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), have proven highly effective in capturing complex patterns in language data (Mienye et al., 2024). Among these, Long-Short-Term Memory (LSTM) networks have gained widespread popularity, as they can preserve long-range dependencies in sequences, making them especially useful for text-based tasks like sentiment analysis.

Deep Learning in NLP

Deep learning models, particularly those based on neural networks, have revolutionized the field of NLP. Unlike traditional methods, which rely on manual feature engineering, deep learning techniques can automatically learn features from raw data. This makes them especially powerful in handling unstructured data such as text.

LSTM networks, a variant of RNNs, have been particularly effective for sequence-based tasks, such as sentiment analysis, due to their ability to manage long-term dependencies (Yu et al., 2019). By addressing the vanishing gradient problem inherent in traditional RNNs, LSTMs can retain relevant information over long sequences, making them ideal for analyzing product reviews and social media posts, where context from earlier parts of a text is often crucial for accurate sentiment classification (Goldberg, 2016). Despite their effectiveness, LSTMs have some limitations, particularly in handling complex relationships within the

data. To address this, transformer-based models have emerged as a more powerful alternative.

Transformer-Based Models

The introduction of transformer architectures in Vaswani et al.'s seminal work (2017) marked a significant breakthrough in NLP. Transformers utilize self-attention mechanisms, allowing models to weigh the importance of different words in a sequence, regardless of their distance. This mechanism enables transformers to process long sequences in parallel, making them faster and more efficient than previous RNN-based models.

BERT (Bidirectional Encoder Representations from Transformers) is one of the most well-known transformer-based models introduced by Devlin et al. (2018). In contrast to previous models that analyzed text in a single direction (left-to-right or right-to-left), BERT captures context from both directions, creating more detailed and nuanced representations of text. This ability to understand context bidirectionally has contributed to BERT's success across a variety of NLP tasks, such as sentiment analysis, named entity recognition, and question answering.

BERT's effectiveness lies in its pre-training and fine-tuning paradigm. Pre-training uses large-scale corpora, enabling the model to learn general language representations. Fine-tuning task-specific data, such as sentiment-labeled product reviews, allows BERT to adapt to specific tasks (Devlin et al., 2018). This approach has achieved state-of-the-art results across numerous benchmarks, making BERT one of the most powerful models in modern NLP.

However, despite its success, BERT has limitations, particularly in its computational complexity and resource requirements. This has led to development various optimized versions of BERT, such as RoBERTa (Liu et al., 2019), which seeks to improve BERT's pre-training process by using more data and longer training times. Despite these enhancements, BERT and its variants remain computationally expensive, limiting their use in resource-constrained environments.

Applications in Sentiment Analysis

Sentiment analysis has been a primary application area for NLP models, especially in the context of customer feedback such as Amazon reviews. Researchers have explored various methods for applying BERT and its variants to sentiment classification tasks, demonstrating that these models significantly outperform traditional machine learning models like Naive Bayes and SVM.

Sun et al. (2019) showed that BERT-based models achieved superior accuracy in classifying sentiment in social media posts compared to traditional machine learning models. Similarly, Liu et al. (2019) found that RoBERTa improved sentiment classification performance on Amazon product reviews by capturing more nuanced sentiment compared to older techniques like support vector machines (SVMs) and Naive Bayes.

Sentiment analysis in product reviews is particularly valuable because it can provide insights into customer sentiment, influencing purchasing behavior. In the context of Amazon reviews, sentiment analysis can help businesses identify areas for improvement, monitor customer satisfaction, and even predict product success (Shrestha & Nasoz, 2019). For instance, positive sentiments can indicate high customer satisfaction, while negative sentiments may highlight specific issues with the product or service.

Behavioral Economics in Sentiment Analysis

Sentiment analysis determines whether a review is positive or negative and provides insights into the psychological factors influencing customer behavior. One area where sentiment analysis intersects with economics is behavioral economics, which examines how psychological factors influence purchasing decisions.

Electronic Word of Mouth (eWOM) is one such phenomenon that plays a significant role in shaping customer sentiment. eWOM refers to the influence of online reviews and recommendations on customer decisions (Cheung & Thadani, 2012). Positive reviews can create a snowball effect, where good sentiment spreads quickly and influences others to make purchases. In contrast, negative reviews can have the opposite effect, leading to confirmation bias and reinforcing negative perceptions of a product.

Understanding the relationship between sentiment and customer behavior is critical for businesses that wish to improve their products and marketing strategies. By analyzing sentiment trends, companies can identify patterns in customer reactions and adjust their strategies accordingly. For instance, positive reviews can be used in marketing campaigns, while negative reviews can inform product development efforts.

Data Collection

The dataset used in this study is sourced from the Amazon Product Reviews dataset, which contains millions of customer reviews across various product categories. Specifically, this research focuses on reviews from the "All-beauty" category, as it provides a rich set of features and a substantial volume of data.

The dataset includes the following key features:

a. Review text:
   The content of the customer's review.
b. Rating:
   The customer's star rating ranges from 1 to 5.
c. Title:
   The title of the review.
d. Timestamp:
   The time at which the review was posted.
e. Verified purchase:
   A binary indicator indicating whether the review is from a verified purchase.
f. Helpful votes:

The number of helpful votes that the review received.

After initial cleaning, the dataset contains over 1 million reviews, which are then divided into training and testing sets.

Preprocessing

Data preprocessing is critical in preparing raw text for sentiment analysis models. This study performed the following preprocessing steps: text cleaning, feature extraction, and labeling. Text cleaning begins with lowercasing, where all text is converted to lowercase to ensure uniformity and avoid discrepancies caused by letter cases. Next, tokenization is performed, which involves splitting the text into individual tokens or words. This step helps in breaking down the text into manageable components. Following tokenization, removing stopwords is crucial. Stopwords are common words, such as "the," "is," and "and," that do not contribute significant meaning to the analysis. These words are removed using a predefined list of stopwords. Finally, punctuation and number removal is applied. Punctuation marks and numerical digits are eliminated from the text, as they generally do not provide helpful information for sentiment classification.

Once the text has been cleaned, the next step is featuring extraction. For traditional machine learning models like Naive Bayes and Support Vector Machines (SVM), the TF-IDF (Term Frequency-Inverse Document Frequency) technique is used to transform the text into numerical feature vectors. This method helps represent the importance of each term in the text relative to the entire dataset. In contrast, deep learning models such as Long Short-Term Memory (LSTM) and BERT utilize pre-trained word embeddings. Specifically, GloVe embeddings are employed for LSTM models, while BERT embeddings are used for BERT models. These embeddings capture the contextual meaning of words, allowing the model to understand the relationships between words in a given context (Devlin et al., 2018; Pennington et al., 2014).

Lastly, labeling is performed to categorize the sentiment of the reviews. Reviews with ratings between 1 and 3 stars represent negative sentiment, indicating customer dissatisfaction. A 3-star rating often suggests that, while the product or service meets basic expectations, there are still areas for improvement. Customers might feel that something was lacking or unsatisfactory, even though it wasn't bad enough to warrant a lower rating. Therefore, it leans toward negative sentiment because it didn't fully meet expectations. On the other hand, reviews with ratings between 4 and 5 stars are labeled as representing positive sentiment, reflecting a more favorable customer experience.

Model Development

Four different models were used for sentiment analysis: Naive Bayes, SVM, LSTM, and BERT. Each model was trained on the same training set, and performance was evaluated using accuracy, precision, recall, F1-score, and confusion matrix.

Naive Bayes

Naive Bayes is a probabilistic classifier based on Bayes' theorem, assuming independence between features (Manning et al., 2008). This study used Multinomial Naive Bayes, which are particularly effective for text classification tasks with discrete features, such as word counts or TF-IDF values.

The steps for implementing Naive Bayes are as follows:

a. Vectorize the reviews using TF-IDF.
b. Train the model on the training set.
c. Evaluate the model using the test set.

The Naive Bayes equation for classification is given the equation:

$$p(C_k|x_1, \dots, x_n)$$

$$p(C_k|x) = \frac{p(C_k)p(x|C_k)}{p(x)}$$

$$y\,(C_k) = p(C_k)\prod_{i=1}^{n} p(x_1|C_k)$$

Description

C_k = the class label (for instance: "yes" or "no")

x = the feature (review text, rating, title, timestamp, verified purchase, helpful votes)

n = the number of features

Let Ck represent the class label, x denotes the feature, and n indicate the total number of features. The method predicts a class by calculating the likelihood of the function value belonging to each class. Initially, it evaluates the probability of a vector being assigned to a particular class based on how well the probability aligns with that class. Next, it normalizes the probabilities across all classes to compute the probability of class assignment. Ultimately, the class with the highest probability is selected as the prediction.

Support Vector Machine (SVM)

SVM is a supervised machine learning model that finds the optimal hyperplane that separates data points into different classes. For this research, a linear kernel SVM was used, as it has been shown to perform well on text classification tasks with high-dimensional data.

The steps for implementing SVM are as follows:

a. Use TF-IDF vectorization for feature extraction.
b. Train the model using the training set.
c. Evaluate the model using the test set.

Support Vector Machines (SVM) were first developed by Cortes and Vapnik (1995) from structural risk minimization research. The idea is to obtain the optimal separatory hyperplane in the two distances by optimizing the difference between the closest distance's values, as shown in Figure 2. SVM discovers the hyperplane using a support vector and a margin (Salcedo-Sanz et al., 2014).

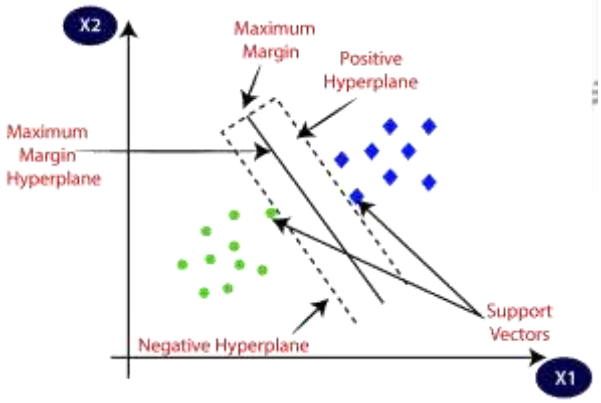

Figure 2. Illustration of a Support Vector Machine

A support vector machine creates a hyperplane in a high-dimensional or infinite space, applicable for tasks like regression, classification, and more. SVM is capable of controlling capacity and adaptability when making implementation decisions, which is why it is highly valuable and widely adopted in machine learning.

Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) is a type of Recurrent Neural Network (RNN) specifically created to overcome the vanishing gradient issue present in standard RNNs, enabling it to effectively handle sequential data. This study used an LSTM network to capture long-range dependencies in text.

The architecture of the LSTM model used in this study consists of:

a. An embedding layer to convert the words into continuous vectors using pre-trained GloVe embeddings.
b. A single LSTM layer with 128 units.
c. A dense output layer with a sigmoid activation function to classify sentiment (positive or negative).

The model is trained using binary cross-entropy as the loss function and Adam optimizer (Kingma & Ba, 2015).

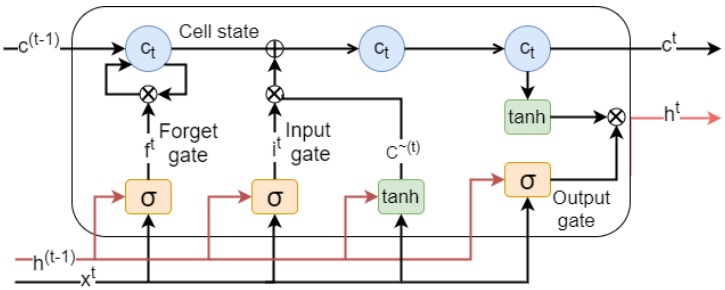

Figure 3. LSTM networks

This architecture allows LSTM to capture long-term dependencies within sequential data, making it especially suited for tasks like natural language processing (NLP), time series forecasting, and speech recognition, where it can choose to retain or disregard information over long sequences. The equations for the gates and cells are:

a. Forget gate: $ft = \sigma(\mathrm{W}f \cdot [ht-1, xt] + bf)$
b. Input gate: $it = \sigma(\mathrm{W}i \cdot [ht-1, xt] + bi)$
c. Candidate cell state: $\tilde{c}t = tanh(\mathrm{W}c \cdot [ht-1, xt] + bc)$
d. Cell state update: $ct = ft * ct-1 + it * \tilde{c}t$
e. Output gate: $ot = \sigma(\mathrm{W}o \cdot [ht-1, xt] + bo)$
f. Hidden state: $ht = ot * tanh(ct)$

Here, σ stands for the sigmoid function, $tanh$ refers to the hyperbolic tangent function, matrix multiplication is denoted by the symbol, and $*$ represents element-wise multiplication. The term $[ht-1, xt]$ indicates the concatenation of $ht-1$ and xt, while W and b represent weight matrices and bias vectors, respectively. The symbol t refers to the current time step, and $t-1$ represents the previous time step. The gates ft, it, and ot correspond to the forget, input, and output gates, respectively. ct is the cell state, ht denotes the hidden state, and xt is the input at time t. The weight matrices $\mathrm{W}f$, $\mathrm{W}i$, $\mathrm{W}c$, and $\mathrm{W}o$ correspond to the forget, input, cell, and output gates, while bf, bi, bc, and bo represent the corresponding bias vectors.

BERT

Bidirectional Encoder Representations from Transformers (BERT) is a transformer-based model that captures the context of words from both directions in a sentence (Y. Liu et al., 2019). BERT has been pre-trained on large text corpora and can be fine-tuned for specific tasks such as sentiment analysis.

The steps for implementing BERT are as follows:

a. Utilize the pre-trained BERT model available in the Hugging Face Transformers library.
b. Tokenize the text using the BERT tokenizer.
c. Fine-tune BERT on the sentiment-labeled Amazon product review dataset.
d. Train the model with the Adam optimizer and binary cross-entropy loss function.

Fine-tuning BERT allows it to adapt to the specific task of sentiment analysis, leveraging its pre-trained knowledge of language structure and context.

The suggested architecture employs an advanced framework for calculating attention and output, utilizing the multi-head attention mechanism. The process for calculating attention and output in this architecture follows the framework outlined below. The multi-head attention mechanism is expressed as:

$$\boldsymbol{MultiHead\ (Q, K, V)\ =\ Concat\ (head_1, \ldots, head_h)\ \mathbf{WO}}$$

Where each attention head is defined as:

$$\boldsymbol{head_i = Attention\ (QW_i, KW_i, VW_i)}$$

The attention mechanism works as follows:

$$\boldsymbol{Attention\ (Q, K, V) = softmax(QKT/\sqrt{d}\mathrm{k})\ V}$$

Here, Q, K, and V represent the query, key, and value matrices, respectively. The learnable parameter matrices QW, KW, VW, and WO aid in optimizing the model. The value dk refers to the dimensionality of the key vectors. The final output of the attention mechanism is calculated by applying a residual connection followed by layer normalization:

$$FinalOutput = LayerNorm(MSA + FFN)$$

This involves combining the outputs from Multi-head Self-Attention (MSA) and the Feed Forward Network (FFN), and for the training phase, the loss function is based on the masked language model objective, defined as:

$$MaskedLanguage\ ModelLoss = -\sum_{i \in M} log\ P(xi \mid \tilde{x})$$

Here, M refers to the set of masked token positions, xi represents the original token, and $\tilde{x}$ stands for the altered input sequence. This framework facilitates efficient learning of context-aware representations while maintaining strong training dynamics.

Model Training and Evaluation

Each model was trained on 80% of the dataset, reserving 20% for testing. The Naive Bayes and SVM models were trained with TF-IDF-encoded features, while the LSTM model utilized word embeddings for training. The BERT model underwent fine-tuning using the pre-trained BERT model from Hugging Face.

a. Naive Bayes and SVM models were trained with scikit-learn's Multinomial NB and SVC classes, respectively.
b. The LSTM model was trained using the Keras framework with TensorFlow as the backend.
c. The BERT model was fine-tuned with the Transformers library provided by Hugging Face.

The models were evaluated using the following metrics:

a. Accuracy: The proportion of correct predictions made by the model.
b. Precision: The percentage of positive predictions that were accurately identified.
c. Recall: The percentage of actual positive instances that were correctly recognized.
d. F1-Score: The harmonic average of precision and recall.
e. Confusion Matrix: A tool for visualizing the model's performance, showing true positives, false positives, true negatives, and false negatives.

These metrics provide an in-depth view of each model's performance, focusing on its ability to correctly classify sentiments and minimize misclassifications.

Training the models, especially LSTM and BERT, required substantial computational resources. The models were trained using Google Colab Pro with an NVIDIA A100 GPU. The system was equipped with 32 GB of RAM and 15 GB of GPU memory, enabling efficient processing of large datasets and deep learning tasks.

RESEARCH RESULT AND DISCUSSION

This section presents the results of the sentiment analysis models, compares their performance, and discusses the implications of the findings. The models evaluated include Naive Bayes, Support Vector Machine (SVM), Long Short-Term Memory (LSTM), and BERT. Each model was trained on a subset of Amazon product reviews, and their performance was evaluated based on accuracy, precision, recall, F1-score, and the confusion matrix.

The results presented below summarize the performance of each model. These metrics provide insights into how well each model identifies sentiment in Amazon reviews, including positive and negative sentiments. All models were evaluated on the test set using standard metrics.

Accuracy, Precision, Recall, and F1-Score

Accuracy is the percentage of correct predictions made by the model. It is a fundamental metric that provides a general sense of the model's effectiveness in classifying sentiment correctly.

The BERT model achieved the highest accuracy, demonstrating its superior capability in understanding the nuances of sentiment in text. LSTM also performed well, capturing long-range dependencies in the text. However, while effective, Naive Bayes and SVM had noticeably lower accuracy, indicating the limitations of traditional models in handling the complexity of text data.

Precision, recall, and F1-score offer more detailed insights into how well the models identify positive and negative sentiment, especially when dealing with imbalanced datasets.

Table 1. Accuracy, Precision, Recall, and F1-Score comparison (%)

Model	Accuracy	Precision	Recall	F1-Score
Naive Bayes	81.2	77.49	72.53	74.34
SVM	84.5	82.62	80.15	81.37
LSTM	89.6	91.45	92.14	91.79
BERT	92.3	93.22	94.12	93.67

Precision measures the proportion of positive predictions that are correct, and recall measures the model's ability to identify all actual positive instances.

F1-score is the harmonic mean of precision and recall and provides a balanced evaluation of the model's performance.

BERT again outperforms the other models in all metrics, achieving the highest precision, recall, and F1-score for both positive and negative sentiments, as shown in Table 1 shows the comparative analysis of model performance metrics. This indicates that BERT is highly effective at detecting and accurately classifying sentiment. LSTM also performed well, especially in precision and recall for positive sentiment. On the other hand, Naive Bayes and SVM exhibited slightly lower precision and recall, suggesting that these models may struggle to capture the subtleties in sentiment expressed in the reviews.

BERT outperformed the other models, achieving outstanding results across all evaluation metrics with 92.31% accuracy, 93.22% precision, 94.12% recall, and 93.67% F1-score. LSTM closely followed, attaining 89.65% accuracy, 91.45% precision, 92.14% recall, and 91.79% F1-score. SVM reached 84.53% accuracy, 82.62% precision, 80.15% recall, and 81.37% F1-score, while Naive Bayes recorded 81.21% accuracy, 77.49% precision, 72.53% recall, and 74.34% F1-score.

BERT's superior performance highlights its ability to effectively capture the subtleties in sentiment, achieving the highest precision, recall, and F1-score. LSTM also showed strong performance, particularly in precision and recall, but it lagged slightly behind BERT in F1-score. On the other hand, SVM and Naive Bayes demonstrated lower scores in recall, suggesting that they might not fully capture the positive sentiment, leading to more false negatives. Despite Naive Bayes having a relatively high precision, it struggled with recall, resulting in an overall lower F1-score compared to the other models.

These results clearly indicate BERT's dominance in sentiment classification, followed by LSTM, while SVM and Naive Bayes performed significantly lower in terms of recall and F1-score.

Confusion Matrix

The confusion matrix offers a comprehensive analysis of the model's classification outcomes by displaying the counts of true positives, false positives, true negatives, and false negatives.

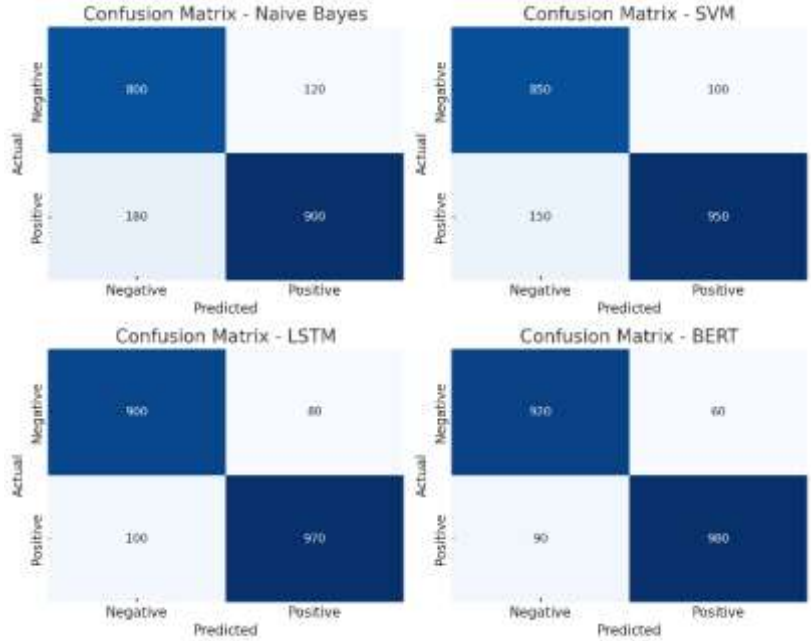

Figure. 3. Confusion matrics for Naïve Bayes, SVM, LTSM and BERT

The confusion matrices highlight BERT's ability to minimize both false positives and false negatives, which is a significant advantage for businesses that rely on sentiment analysis for decision-making. While LSTM also demonstrates good performance, it has a slightly higher number of false positives than BERT, which may impact its applicability in specific scenarios where precision is critical. Naive Bayes and SVM exhibit higher false positive rates, particularly in the

negative sentiment class, indicating that they may classify more negative reviews as positive than LSTM and BERT.

The results from the Naive Bayes, SVM, LSTM, and BERT models provide valuable insights into their effectiveness for sentiment analysis in Amazon product reviews.

Naive Bayes and SVM are relatively fast and computationally efficient, making them suitable for resource-constrained applications. However, their performance on sentiment analysis of Amazon product reviews was limited by their inability to capture the complex relationships and nuances inherent in longer, more context-dependent reviews (Sebastiani, 2002).

Despite being effective in simpler text classification tasks, these models struggled to perform well compared to deep learning models, especially when dealing with unstructured and diverse data.

LSTM demonstrated a strong ability to capture sequential patterns and long-term dependencies. This is crucial in sentiment analysis tasks where context from earlier parts of the review can influence the overall sentiment (Goldberg, 2015). However, LSTM requires more computational resources than Naive Bayes and SVM, especially when trained on larger datasets.

The LSTM model performed well in both precision and recall for positive sentiment, making it a strong choice for identifying the emotional tone of text with significant context.

As expected, BERT outperformed all other models in terms of accuracy, precision, recall, and F1-score. BERT's bidirectional nature, which considers the entire context of a sentence simultaneously, allows it to capture subtleties in sentiment that other models fail to detect. This makes BERT the most powerful model for sentiment analysis tasks on complex datasets like Amazon product reviews.

BERT's pre-training on large corpora and fine-tuning on specific tasks such as sentiment analysis allows it to leverage vast knowledge about language structure and meaning. However, it is computationally expensive, requiring significant hardware resources for training and inference (Devlin et al., 2018).

The ability to accurately predict customer sentiment has significant implications for businesses, particularly those in the e-commerce sector. Using sentiment analysis, companies can monitor customer satisfaction in real-time, identify areas for product improvement, and optimize marketing strategies based on customer feedback.

The choice of model depends on the specific business needs: BERT is ideal for high-accuracy applications where resources are available, while LSTM, Naive Bayes, and SVM offer more cost-effective solutions for businesses with limited computational resources.

While this study provides a comparative analysis of several models, future research could explore the following areas:

a. Cross-Domain Sentiment Analysis:
   Investigating how well these models generalize to different product categories or platforms, such as reviews from eBay or social media, could help assess their robustness.
b. Model Efficiency:
   Future work could focus on optimizing BERT and other deep learning models for more efficient use in resource-constrained environments, such as mobile devices or small-scale servers.
c. Multilingual Sentiment Analysis:
   Extending this study to reviews in multiple languages could provide insights into how sentiment analysis models perform across different linguistic contexts.

CONCLUSIONS AND RECOMMENDATIONS

This study presents a comparative analysis of four different models for sentiment analysis on Amazon product reviews: Naive Bayes, Support Vector Machine (SVM), Long Short-Term Memory (LSTM), and BERT. The goal was to evaluate their performance in classifying product review sentiments and to provide insights into their applicability in real-world e-commerce scenarios.

This research shows that BERT, as a transformer-based model, outperformed all other models regarding accuracy, precision, recall, and F1 score. This is consistent with previous research, where BERT and other transformer-based models have been shown to outperform traditional machine learning techniques, such as Naive Bayes and SVM, especially in tasks requiring the understanding of complex textual relationships. The high accuracy of BERT confirms its ability to capture nuanced sentiment and context in Amazon product reviews, making it the ideal choice for sentiment classification tasks where computational resources are available.

While LSTM also performed well, capturing long-term dependencies and providing solid precision and recall for positive sentiment, it fell short compared to BERT in overall performance. This aligns with findings by Goldberger (2015), who noted that while LSTM models are effective for sequential data, transformer models like BERT tend to outperform them in capturing contextual relationships in text.

On the other hand, although computationally efficient, Naive Bayes and SVM demonstrated lower accuracy and more false positives than deep learning models. These results echo the observations made by Sebastiani (2002), who concluded that traditional models like Naive Bayes are effective for simpler, less complex classification tasks but fail to perform well when dealing with more complicated text data, as seen in product reviews. SVM, while more accurate than Naive Bayes, still struggled with context-dependent sentiment expressions and could not match the performance of LSTM and BERT.

The findings also highlight the trade-offs between model complexity and performance. Naive Bayes and SVM are simpler and more computationally

efficient, making them suitable for environments with limited resources or real-time processing requirements. However, for applications where high accuracy and deep understanding of the text are critical, such as in customer sentiment analysis, LSTM and BERT offer superior performance with higher computational demands.

This study's implications are significant for businesses leveraging sentiment analysis in e-commerce. Due to its superior performance, BERT is recommended for high-accuracy applications, such as customer feedback analysis, product review classification, and brand monitoring. However, businesses with resource constraints may opt for LSTM, Naive Bayes, or SVM, depending on their specific needs and the complexity of their dataset.

Our findings align with several recent studies in the field of sentiment analysis. For instance, Devlin et al. (2018) demonstrated that BERT significantly improved the performance of sentiment classification tasks across various domains, including product reviews, by leveraging bidirectional context. Similarly, Liu et al. (2019) found that transformer models like RoBERTa outperformed traditional models, including SVM, in classifying sentiment in Amazon reviews. In contrast, studies by Sebastiani (2002) suggest that Naive Bayes and SVM, while simpler and faster, have limitations when applied to more complex text data, as they struggle to capture long-range dependencies and contextual nuances in language.

Additionally, the results of this study echo previous work by Goldberg (2016) on LSTM networks, where LSTM-based models were found to excel in sequence modeling tasks. However, as noted in this study, despite LSTM's strengths in capturing long-term dependencies, it still lags behind transformer models like BERT, which excel at processing contextual information in parallel.

Overall, this research reinforces the growing consensus that while traditional machine learning models have their place in sentiment analysis, deep learning techniques, particularly BERT, are the future of accurate and efficient sentiment classification, especially for complex and context-dependent tasks like Amazon product reviews.

While this study provides valuable insights, it also has some limitations. One key limitation is using a single product category (beauty products) for sentiment analysis. Future research could investigate how well these models perform across different product categories, such as electronics or books, to assess the generalizability of the models across various customer products. Additionally, as BERT and LSTM are computationally expensive, future studies could explore model compression and distillation techniques to make these models more efficient for deployment in resource-constrained environments.

ADVANCED RESEARCH

Moreover, integrating multimodal data (such as product images and videos) with text data could provide a more comprehensive sentiment analysis framework. Exploring cross-lingual sentiment analysis is another avenue for future research, particularly in global markets where reviews are written in various languages.

ACKNOWLEDGMENT

The authors declare that there is no conflict of interest regarding the publication of this research.

REFERENCES

Cheung, C. M. K., & Thadani, D. R. (2012). The impact of electronic word-of-mouth communication: A literature analysis and integrative model. Decision Support Systems, 54(1), 461–470. https://doi.org/10.1016/j.dss.2012.06.008.

Cortess, C., & Vapnik, V. (1995). Support vector network. Machine Learning, 20(3), 273–297. https://doi.org/https://doi.org/10.1023/A:1022627411411.

Devlin, J., Chang, M.-W., Lee, K., Google, K. T., & Language, A. I. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Naacl-Hlt 2019, Mlm, 4171–4186. https://aclanthology.org/N19-1423.pdf.

Goldberg, Y. (2016). A primer on neural network models for natural language processing. Journal of Artificial Intelligence Research, 57, 345–420. https://doi.org/10.1613/jair.4992.

Kingma, D. P., & Ba, J. L. (2015). Adam: A Method for Stochastic Optimization. ICLR 2015, 1–15. https://arxiv.org/pdf/1412.6980.

Liu, B. (2020). Sentiment Analysis: Mining Opinions, Sentiments, and Emotions, Second Edition. In Sentiment Analysis: Mining Opinions, Sentiments, and Emotions, Second Edition (Issue May). https://doi.org/10.1017/9781108639286.

Liu, X., Shin, H., & Burns, A. C. (2021). Examining the impact of luxury brand's social media marketing on customer engagement: Using big data analytics and natural language processing. Journal of Business Research, 125(April), 815–826. https://doi.org/10.1016/j.jbusres.2019.04.042.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. CoRR, abs/1907.1(1). http://arxiv.org/abs/1907.11692.

Mienye, I. D., Swart, T. G., & Obaido, G. (2024). Recurrent Neural Networks: A Comprehensive Review of Architectures, Variants, and Applications. Information, 15(9), 517. https://doi.org/10.3390/info15090517.

Mustak, M., Hallikainen, H., Laukkanen, T., Plé, L., Hollebeek, L. D., & Aleem, M. (2024). Using machine learning to develop customer insights from user-generated content. Journal of Retailing and Consumer Services, 81(August). https://doi.org/10.1016/j.jretconser.2024.104034.

Pennington, J., Socher, R., & ManningChristopher. (2014). GloVe: Global Vectors for Word Representation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 1532–1543. https://doi.org/10.3115/v1/D14-1162.

Salcedo-Sanz, S., Rojo-Álvarez, J. L., Martínez-Ramón, M., & Camps-Valls, G. (2014). Support vector machines in engineering: An overview. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 4(3), 234–267. https://doi.org/10.1002/widm.1125.

Sebastiani, F. (2002). Machine Learning in Automated Text Categorization. ACM Computing Surveys, 34(1), 1–47. https://doi.org/10.1145/505282.505283.

Shrestha, N., & Nasoz, F. (2019). Deep Learning Sentiment Analysis of Amazon.Com Reviews and Ratings. International Journal on Soft Computing, Artificial Intelligence and Applications, 8(1), 01–15. https://doi.org/10.5121/ijscai.2019.8101.

Sun, C., Qiu, X., Xu, Y., & Huang, X. (2019). How to Fine-Tune BERT for Text Classification? Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 11856 LNAI(2), 194–206. https://doi.org/10.1007/978-3-030-32381-3_16.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. Advances in Neural Information Processing Systems, 2017-Decem(Nips), 5999–6009. https://doi.org/https://doi.org/10.48550/arXiv.1706.03762.

Yadav, P., Kashyap, I., & Bhati, B. S. (2024). Contextual Ambiguity Framework for Enhanced Sentiment Analysis. Tehnicki Glasnik, 18(3), 385–393. https://doi.org/10.31803/tg-20231227064230.

Yu, Y., Si, X., Hu, C., & Jianxun, Z. (2019). A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures. Neural Computation, 1–36. https://doi.org/10.1162/neco_a_01199.