Real-time Sentiment Analysis of E-commerce Product Reviews

Dau Dinh Quang Anh[1,2,3], Kieu Xuan Dieu Huong[1,2,3], Nguyen Nhat Thuong[1,2,3], and Le Quang Hoa[1,2,3]

[1] University of Information Technology, Ho Chi Minh City, Vietnam
[2] Vietnam National University, Ho Chi Minh City, Vietnam
[3] {20521059,20521381,20522000,20521331}@gm.uit.edu.vn

Abstract. This paper focuses on sentiment analysis of customer reviews in the context of e-commerce platforms. With the exponential growth of online shopping and customer reviews, sentiment analysis has become a crucial task to extract valuable insights from textual data. The study utilizes various machine learning and deep learning models, including BERT, LSTM, MLP, and traditional algorithms like Random Forest, to classify sentiments effectively. The pretrained BERT model achieves the highest F1-Score of 83%, outperforming other models. Additionally, the research employs the Apache Spark framework for distributed data processing to handle large-scale, continuous data from social media platforms

Keywords: Big Data · NLP · Sentiment Analysis · SparkNLP · Opinion Mining.

1 Introduction

The rapid growth of e-commerce platforms and the increasing popularity of online shopping, customer reviews have become a vital source of information for potential buyers. Customers often share their opinions, experiences, and sentiments about products in the form of written reviews. Analyzing these vast amounts of textual data manually is a challenging task, but advancements in Natural Language Processing (NLP) and Machine Learning have made it possible to automate this process.

Sentiment analysis, also known as opinion mining, involves determining the emotional tone expressed in a piece of text, such as whether it is positive, negative, or neutral towards a particular subject. To process sentiment analysis effectively, the text data undergoes preprocessing to remove irrelevant information, like punctuation and stopwords. Additionally, techniques like text normalization, stemming, and lemmatization are applied for consistency in word representation. Word embeddings play a crucial role, capturing the semantic meaning of words and phrases in a continuous vector space, enabling machines to understand relationships between words and their contextual relevance. These embeddings form the foundation for sentiment analysis models, which use machine learning algorithms to classify sentiment in real-time product reviews, social media posts, and other textual data. However, applying algorithms to tackle sentiment analysis on today's social media platforms, with billions of users and massive data volumes, poses a significant challenge. Capturing customer feedback and sentiments about products in a continuous and real-time manner is crucial to gaining valuable insights. Therefore, we have utilized the Apache Spark framework to address the sentiment analysis task on large-scale, online, and continuous data. Apache Spark enables distributed data processing, allowing us to apply sentiment analysis algorithms effectively and efficiently in the dynamic and ever-changing environment of social media platforms.

Throughout the experimentation process, the model yielded high accuracy when predicting online and continuous comments. Moreover, the paper also presents a web-based demo implementation. The paper showcases the utilization of Apache Spark and machine learning techniques, alongside deep learning models, for sentiment analysis on large-scale and continuous data. The study holds promising prospects for further development to enhance the results and provide even higher accuracy for real-world applications.

In this paper, we focus on introducing the utilization of traditional machine learning algorithms and several deep learning techniques within Spark for sentiment analysis. In Section 2, we will present related works in the field. In Section 3, we will introduce the dataset used in this study. In Section 4, we will describe the methods employed for data preprocessing and model training. The experimentation process will be detailed in Sections 5 and 6, analyzing the obtained results. Finally, Section 7 will conclude the paper and outline future directions for development.

2 Related Work

Research on sentiment analysis in natural language processing (NLP) has garnered significant attention from the research community in recent years. Several relevant studies have been conducted in this field, highlighting different approaches and applications for sentiment analysis.

One such study conducted by Prajval [3] focuses on the common task of sentiment analysis in natural language processing. Sentiment analysis is a natural language processing technique used to determine whether data has a positive, negative, or neutral sentiment. The paper explores various methods and classifiers employed in the sentiment analysis task. Additionally, the paper "SANA: Sentiment analysis on newspapers comments in Algeria" [7] discusses sentiment analysis or opinion mining within comments on Algerian newspaper websites. The research centers on creating a labeled dataset, consisting of comments from three different Algerian newspapers, annotated by native Arabic-speaking Algerians. This dataset is then utilized to build a classification model for sentiment analysis of the comments.These studies provide valuable insights into sentiment analysis techniques, methods, and applications in the context of natural language processing. Building upon this existing research, our paper aims to further contribute to the field by exploring novel approaches and solutions for sentiment analysis in NLP.

3 Dataset

The dataset used in this project comprises Amazon product reviews, which is a subset of the larger Amazon Product review dataset. It is conveniently stored in the TensorFlow database and can be easily loaded using the 'tfds' API from TensorFlow. The dataset contains more than 100,000 rows and includes various columns ranging from Product ID to reviews, headings, and star ratings provided by the customers. Since our focus is on analyzing the reviews and their corresponding ratings, we will discard the other feature columns. For this project, we work with a subset of 20,000 rows randomly selected from the dataset. Table 1 contains examples from the original dataset.

Table 1. Examples from the original dataset.

No	Comment	Star Rating
1	*Does not work.*	1
2	*This is a great wiring kit i used it to set up..*	4
3	*It works great so much faster than USB charger...*	5
4	*This product was purchased to hold a monitor o...*	3

To fit the task, we perform data transformation. Specifically, for the customer comment sentiment analysis task, we label comments with a rating of 3 stars or higher as "positive" and label comments with a rating below 3 stars as "negative".

Table 2. Examples after preprocess the original dataset.

No	Comment	Sentiment
1	*Does not work.*	Negative
2	*This is a great wiring kit i used it to set up..*	Positive
3	*It works great so much faster than USB charger...*	Positive
4	*This product was purchased to hold a monitor o...*	Positive

Table 3 presents the label statistics of the dataset for the task of sentiment analysis based on comments. There is a significant imbalance between the "positive" and "negative" labels. The majority of the dataset consists of comments labeled as "positive."

Table 3. Label Quantity

Set	Label	Qty
Training	Positve	10621
	Negative	3379
Test	Positive	4596
	Negative	1404

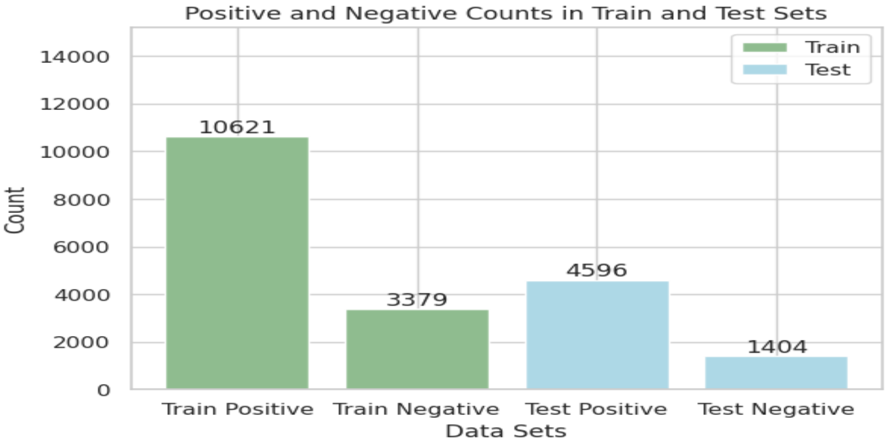


Fig. 1. Visualization label quantity.

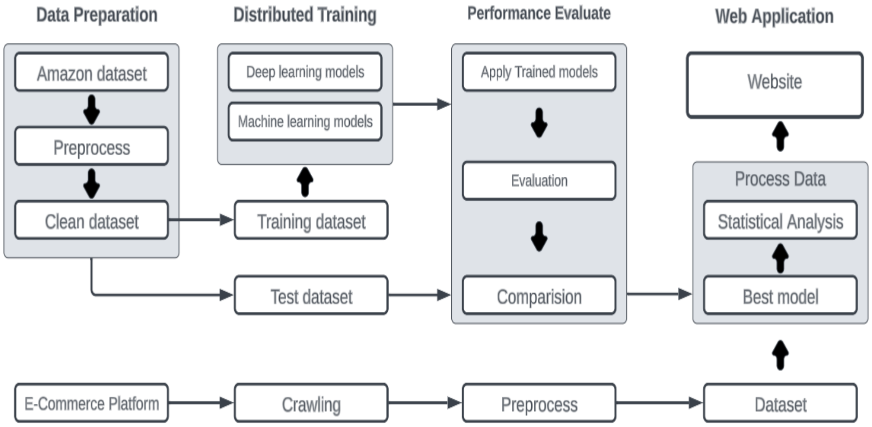


Fig. 2. Approach to the sentiment analysis.

4 Method

The process of sentiment analysis for customer comments on products involves several key steps. Firstly, the Amazon product reviews dataset is utilized, and through careful data preprocessing, the information is cleaned and prepared for analysis. Next, various models are trained using this preprocessed data to perform sentiment analysis effectively. These models are then rigorously evaluated to identify the one with the best performance. Subsequently, customer comments from different e-commerce platforms are crawled, and the collected data undergoes thorough cleaning for further analysis. Using the best-performing model, sentiment predictions are made on the crawled data. The predicted sentiments are then subjected to statistical analysis, which provides valuable insights into customer sentiments across different products. Ultimately, these analyzed results are presented on the website, enabling users to access and gain comprehensive insights into the sentiments expressed by customers regarding various products.

4.1 Data preprocessing

In natural language processing (NLP) in general, and specifically in English language processing, data preprocessing is a crucial and indispensable stage. The first step in data processing is data preprocessing, which plays a crucial role in model building and achieving accurate and effective results in classification, prediction, or information extraction tasks from natural language text. Data preprocessing is particularly important in the task of sentiment analysis in sentence-level comments. Standardizing the data and removing irrelevant components are essential to optimize the data before training the sentiment analysis model. This step helps the model better understand and learn information from the data, leading to improved accuracy and effectiveness of the sentiment classification model. The data preprocessing steps carried out in the sentiment analysis of sentence-level comments include:

Tokenizer: Tokenization is the process of breaking down the sentence-level comments into individual words or tokens. It is a critical step that transforms the raw text into a structured format, making it easier for further analysis. For example, the sentence ”I love this product!” would be tokenized into the following tokens: [”I”, ”love”, ”this”, ”product”, ”!”].

CountVectorizer: After tokenization, the CountVectorizer is applied to convert the tokenized words into numerical vectors. It creates a bag-of-words representation, where each entry in the vector corresponds to the frequency of a word in the sentence-level comments. This step captures the occurrence of each word and prepares the data for the subsequent modeling steps.

IDF (Inverse Document Frequency): IDF is a weighting scheme that measures the importance of each word in the entire dataset. It assigns higher weights to words that appear less frequently across all sentence-level comments and lower weights to common words that appear in many comments. This is done to emphasize the significance of words that are unique to certain comments and potentially carry more sentiment information.

N-gram: N-grams are contiguous sequences of N words. In sentiment analysis, incorporating n-grams (bi-grams, tri-grams, etc.) provides a more comprehensive understanding of the context and sentiment expressed in the comments. By considering sequences of words, the model can capture dependencies between adjacent words and gain insights into how certain combinations of words contribute to sentiment.

Hashing TF: Hashing Term Frequency (TF) is a technique that maps words to fixed-size vectors using a hashing function. This process helps manage memory and computational resources when dealing with a large vocabulary. The hashing function converts words into numeric values, and the resulting vectors are used to represent the frequency of each word in the sentence-level comments.

By performing these data preprocessing steps, the raw text data is transformed into a numerical representation suitable for training a sentiment analysis model. Each step contributes to enhancing the model's ability to comprehend and learn from the data, leading to improved accuracy and effectiveness in sentiment classification. The optimized data is then used to build a machine learning model capable of predicting sentiment in sentence-level comments accurately.

4.2 Word embedding

Word embedding is a method for representing words as numerical vectors in a multi-dimensional space, which helps computers understand and process natural language more effectively. One of the popular and powerful word embeddings is BERT (Bidirectional Encoder Representations from Transformers).

BERT is a pre-trained language model that is trained on a large amount of text data from the Internet. This type of representation allows BERT to understand the context and dependencies between words in a sentence, including n-grams and surrounding context. This makes BERT a powerful and versatile word embedding model.

BERT operates through the Transformer model, a neural network architecture widely used in NLP. The Transformer model can process words in a sentence simultaneously, capturing complex relationships between words. This enables BERT to learn how to represent words in a meaningful and contextual manner.

After training, BERT generates numerical representation vectors for each word in its vocabulary. As a result, words with similar meanings or close relationships will have vectors that are close to each other in the multi-dimensional space. This allows BERT to understand and predict the semantics of words in the context of a sentence or paragraph.

BERT embedding is used in various NLP applications, such as text classification, information extraction, machine translation, and text summarization. With its ability to understand context and relationships between words, BERT embedding provides a powerful means of representing natural language and improves the performance of NLP tasks.

4.3 Traditional machine learning model

Random Forest : Random Forest is a traditional machine learning algorithm used in sentiment analysis and other classification tasks based on input data. It operates by building multiple decision trees and combining their results to make predictions for new texts. The decision trees are trained on random subsets of the data and possess flexible characteristics, which help to avoid overfitting and improve the accuracy and performance of the sentiment classification model in natural language processing (NLP) tasks. Additionally, Random Forest can handle missing data and can handle data with many variables, making it a versatile tool for sentiment analysis and other NLP tasks. Its ability to handle complex data and make accurate predictions has made it a popular choice for sentiment analysis and other classification tasks.

4.4 Deep learning model

BERT(Bidirectional Encoder Representations from Transformers): is a breakthrough language representation model developed by Google Research in 2018. It leverages the Transformer architecture to learn contextual embeddings for words, capturing the bidirectional context in a sentence. Unlike traditional language models that process text sequentially, BERT reads the entire sentence in both directions, allowing it to better understand the context and meaning of each word. BERT's pre-training process involves learning from a large corpus of text, which enables it to capture rich and contextual language representations. After pre-training, BERT can be fine-tuned on specific downstream tasks, such as sentiment analysis, question-answering, and natural language understanding, leading to state-of-the-art performance on a wide range of NLP tasks. Its ability to capture deep contextual information and transfer knowledge to various tasks has made BERT a foundational model in the field of natural language processing.The BERT model can learn word context through two training strategies:

Masked Language Model (MLM): Before inputting a sequence into the BERT model, 15 percent of the words in the sequence are replaced with the token "[MASK]". The model is then trained to predict the word replaced by "[MASK]" based on the context of the surrounding words that were not masked.

Next Sentence Prediction (NSP): Next Sentence Prediction (NSP): In this strategy, the model uses a pair of sentences as input and predicts whether the second sentence is the actual next sentence following the first one. During training, the input data contains 50 percent sentence pairs where the second sentence is indeed the consecutive sentence to the first one, and the remaining 50 percent randomly selects the second sentence from the dataset.

LSTM(Long Short-Term Memory): LSTM, short for Long Short-Term Memory, is a type of recurrent neural network (RNN) architecture designed to address the vanishing gradient problem. It introduces specialized memory cells that allow information to be stored over long periods, enabling the model to retain important information from earlier time steps. This unique memory mechanism enables LSTMs to effectively capture long-range dependencies in sequential data, making them highly suitable for tasks involving time series data, natural language processing, and various sequential data analysis tasks. LSTM's ability to handle long-term dependencies has made it a popular choice for various applications in deep learning, particularly in tasks requiring sequential modeling and context preservation.

MLP: MLP, or Multi-Layer Perceptron, is a fundamental neural network architecture widely used in machine learning and deep learning applications. Composed of multiple layers of interconnected neurons, it can handle complex and non-linear relationships in data, making it suitable for various tasks such as classification, regression, and pattern recognition. Each neuron in an MLP layer processes input data, and the model learns to adjust the weights through backpropagation during training to minimize prediction errors. Its feedforward nature ensures efficient information flow from input to output without feedback loops. MLP's versatility and ability to learn from large and diverse datasets have made it a cornerstone in modern artificial intelligence, enabling powerful solutions across a wide range of real-world applications.

4.5 Evaluation metric

We use evaluation metrics F1-score to assess the classification performance of models, where TP stands for true positive, TN for true negative, FP for false positive, and FN for false negative, as defined in the following equations:

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

5 Implementation

With BERT, we use two methods for sentiment analysis. The first method is to train our own model. We utilize BERT embeddings along with the provided `ClassifierDL` (a class used to build classification models using deep learning) from the `sparknlp.annotator` package. This approach enables us to perform sentiment classification based on data represented with embeddings from deep language models like BERT. The BERT model we use is `bert_smal_L2_128`,

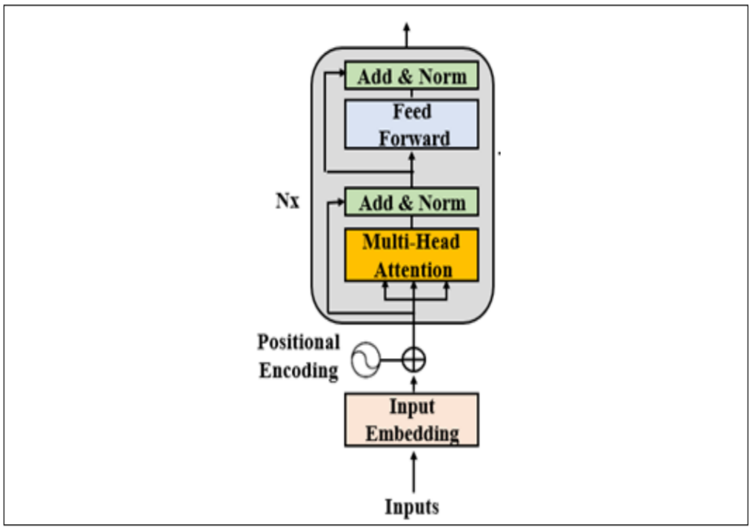


Fig. 3. Encoder layer.

which consists of 2 encoder layers and 128-dimensional vector space. The parameters for this method are as follows: MaxEpochs: 125, Learning Rate: 0.007. ‘The second method involves using a pretrained model and modifying some parameters, such as the threshold (threshold for binary classification) set to 0.5, MaxSequenceLength set to 128, and BatchSize set to 32. The architecture of BERT show in ig 2.

With LSTM, we use an MLP model to create an architecture similar to LSTM in Spark’s Word2Vec to generate embeddings vectors with a vector size of 100 and set the parameters as follows: layers = [100, 64, 32, 3], maxIter = 100, and blockSize = 1000.

With the MLP model using Word2Vec embeddings as the vector embedding, the architecture consists of an input layer with 100 neurons, followed by two hidden layers with 65 and 32 neurons, respectively, and an output layer with 2 neurons. A dropout rate of 0.2 will be applied after each hidden layer.

With the traditional machine learning model, RandomForest, we use different approaches to represent the input data and train the model. The models we utilize are combinations of:

6 Result

BERT (Pretrained) achieved the highest F1-Score of 83%. This result indicates that the pretrained BERT model, which leverages deep learning techniques, outperforms the other models in capturing complex contextual information and patterns in the text data. BERT (Train Using Classifier Approach) obtained an F1-Score of 82%, which is slightly lower than the pretrained BERT model.

However, it still showcases the effectiveness of fine-tuning BERT for the specific sentiment analysis task. LSTM and MLP achieved F1-Scores of 82.59% and 77.4%, respectively. These deep learning models demonstrate competitive performance compared to traditional machine learning models. Among the traditional machine learning models, the ensemble method CV + ngram + IDF + Assembler + Random Forest achieved an F1-Score of 75%, which is higher than HashingTF + IDF + Random Forest and CV + IDF + Random Forest (F1-Scores of 62.84% and 63.09%, respectively). This suggests that combining features through the ensemble approach and incorporating n-gram features can improve the performance of the RandomForest

Table 4. Model Comparison

Model	**Train - Test**	**F1-Score(%)**
BERT (Pretrained)	7 - 3	**83**
BERT (Train Using Classifier Approach)	7 - 3	82
LSTM	7 - 3	82.59
MLP	7 - 3	77.4
HashingTF + IDF + Random Forest	7 - 3	62.84
CV + ngram + IDF + Assembler + Random Forest	7 - 3	75
CV + IDF + Random Forest	7 - 3	63.09

7 Application

7.1 Software requirement

Vision The application aims to provide a solution for automatically analyzing real-time customer comments to gain a competitive advantage for businesses using the system.

Goal (a) Customer experience (b) Continuous improvement (c) Focus on execution (d) Simplified process.

Design Fig 4 is a web application interface that allows users to enter a URL of an e-commerce page to analyze the sentiment of comments. The application allows users to input the Shopee shop's link to classify the sentiment of each comment within it.

7.2 Technology

Flask is an open-source Python framework used for building interactive data visualization web applications. It was developed and released in the middle of 2017.

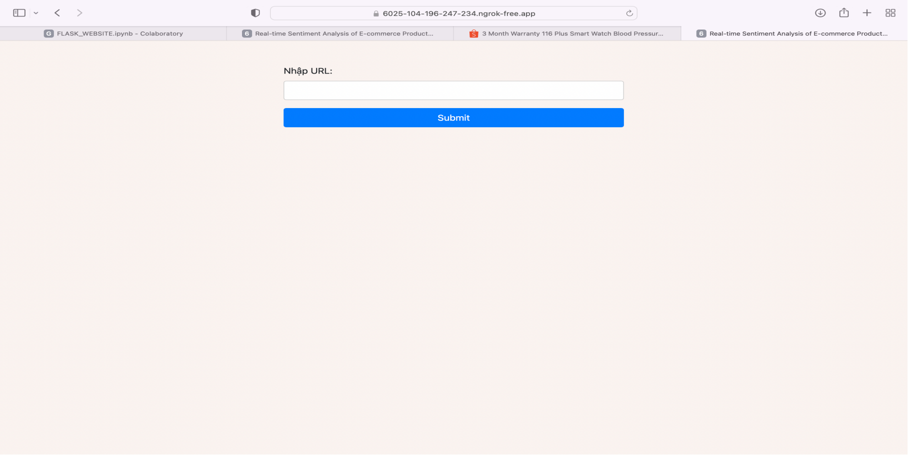
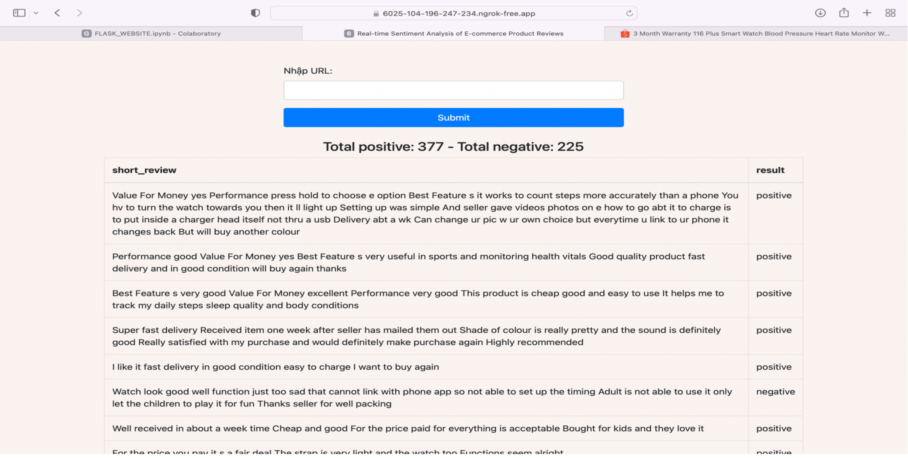

6025-104-196-247-234.ngrok-free.app
FLASK_WEBSITE.ipynb - Colaboratory
Real-time Sentiment Analysis of E-commerce Product...
3 Month Warranty 116 Plus Smart Watch Blood Pressur...
Real-time Sentiment Analysis of E-commerce Product...
Nhập URL:
Submit

Fig. 4. Interface of Website.


6025-104-196-247-234.ngrok-free.app
FLASK_WEBSITE.ipynb - Colaboratory
Real-time Sentiment Analysis of E-commerce Product Reviews
3 Month Warranty 116 Plus Smart Watch Blood Pressure Heart Rate Monitor W...
Nhập URL:
Submit

Total positive: 377 - Total negative: 225

short_review	result
Value For Money yes Performance press hold to choose e option Best Feature s it works to count steps more accurately than a phone You hv to turn the watch towards you then it ll light up Setting up was simple And seller gave videos photos on e how to go abt it to charge is to put inside a charger head itself not thru a usb Delivery abt a wk Can change ur pic w ur own choice but everytime u link to ur phone it changes back But will buy another colour	positive
Performance good Value For Money yes Best Feature s very useful in sports and monitoring health vitals Good quality product fast delivery and in good condition will buy again thanks	positive
Best Feature s very good Value For Money excellent Performance very good This product is cheap good and easy to use It helps me to track my daily steps sleep quality and body conditions	positive
Super fast delivery Received item one week after seller has mailed them out Shade of colour is really pretty and the sound is definitely good Really satisfied with my purchase and would definitely make purchase again Highly recommended	positive
I like it fast delivery in good condition easy to charge I want to buy again	positive
Watch look good well function just too sad that cannot link with phone app so not able to set up the timing Adult is not able to use it only let the children to play it for fun Thanks seller for well packing	negative
Well received in about a week time Cheap and good For the price paid for everything is acceptable Bought for kids and they love it	positive
For the price you pay it s a fair deal The strap is very light and the watch too Functions seem alright	positive

Fig. 5. The result after running.

8 Conclusion and future work

The research paper that focuses on sentiment analysis (opinion mining) of e-commerce product reviews. Sentiment analysis involves determining whether a piece of text expresses a positive or negative toward a specific subject, such as a product. The paper explores the use of Natural Language Processing (NLP) and Machine Learning techniques, particularly using Apache Spark. The paper compares the performance of traditional machine learning algorithms like Random Forest with deep learning models like BERT, LSTM (Long Short-Term Memory), and MLP (Multi-Layer Perceptron). Various models were evaluated for their performance in different natural language processing tasks, and their accuracy was measured using the F1-Score metric. Among the models, BERT (Pretrained) demonstrated the highest accuracy with an impressive F1-Score of 83%. Not far behind, the BERT model trained using a Classifier Approach achieved a commendable F1-Score of 82%. The LSTM model also performed remarkably well, obtaining an F1-Score of 82.59%. The MLP model showed a respectable accuracy with an F1-Score of 77.4%. On the other hand, traditional machine learning models, such as HashingTF + IDF + Random Forest and CV + IDF + Random Forest, exhibited lower accuracies with F1-Scores of 62.84% and 63%, respectively. The most competitive traditional machine learning model was CV + ngram + IDF + Assembler + Random Forest, which achieved an F1-Score of 75%. The results show that BERT achieved the highest F1-Score (a measure of model performance) of 83%, outperforming other models.

In the future, we plan to explore various libraries for building models and conduct experiments with different embedding sets. This is aimed at expanding the scope of our research and discovering new potentials in the field of natural language processing. By exploring new libraries and embeddings, we hope to uncover improvements and offer novel solutions for sentiment analysis, text classification, and other NLP tasks.

References

1. Jelodar, H., Wang, Y., Orji, R., Huang, D., Zhang, Y. (2021). Sentiment Analysis in Social Media. Data Science and Engineering, 6(2), 79-96.
2. Jin, W., Ho, H. H., Srihari, R. K. (2020). Sentiment Analysis on Social Media. Information Sciences, 505, 525-541.
3. Comparative study of various approaches, applications and classifiers for sentiment analysis
4. Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Céspedes, Steve Yuan, Chris Tar, et al. Universal sentence encoder. arXiv preprint arXiv:1803.11175, 2018.
5. Akhtar, M. S., Rehman, A., Ullah, S., Khan, A. U. (2019). Aspect Based Sentiment Analysis: A Survey. Knowledge-Based Systems, 161, 206-220.
6. Sharma, S., Joshi, S., Modi, K. B. (2020). Sentiment Analysis in Indian Languages: A Survey. In Proceedings of the International Conference on Inventive Computation Technologies (ICICT) (pp. 175-179).

7. SANA: Sentiment analysis on newspapers comments in Algeria
8. Yadav, P. K., Yadav, D. (2020). Sentiment Analysis of Product Reviews: A Comprehensive Review. Journal of Ambient Intelligence and Humanized Computing, 11(5), 2061-2086.
9. Liu, B. (2012). Sentiment Analysis and Opinion Mining. Synthesis Lectures on Human Language Technologies, 5(1), 1-167.