International Journal of Engineering Trends and Technology
ISSN: 2231–5381 / https://doi.org/10.14445/22315381/IJETT-V71I11P222

Volume 71 Issue 11, 208-217, November 2023
© 2023 Seventh Sense Research Group®

Original Article

OpExBERT: Opinion EXTRACTION and Classification of Reviews using BERT Model

Ankur Ratmele[1], Ramesh Thakur[2], Archana Thakur[3]

[1]*Computer Engineering, IET-DAVV Indore, Madhya Pradesh, India.*
[2]*IIPS, DAVV Indore, Madhya Pradesh, India.*
[3]*SCSIT, DAVV Indore, Madhya Pradesh, India.*

[1]*Corresponding Author : ankur22ratmele@gmail.com*

Received: 12 July 2023 Revised: 02 October 2023 Accepted: 10 October 2023 Published: 04 November 2023

Abstract *- Sellers and merchants have started asking customers to comment on the products at online marketplaces in recent years. It might be really difficult for a potential buyer to decide whether to buy a product after analyzing the vast amounts of evaluations. This research introduces a solution for tackling this problem, employing a hierarchical attention network approach. In this proposed framework, the initial step involves preprocessing the Amazon Smartphone Review dataset using natural language processing techniques. Subsequently, the BERT (Bidirectional Encoder Representations from Transformers) model is applied to make word vector illustrations of the reviews, capturing their contextual meaning. Developed by Google AI Language, BERT is a deep learning model explicitly designed to pre-train text in an unsupervised manner by considering context from both the left and right directions across all its layers. This is in contrast to previous language representation models that were unidirectional. The pre-trained BERT model can be fine-tuned with just one output layer for different tasks related to language understanding, such as opinion mining and question answering. This allows efficient adaptation to specific tasks without modifying the model's underlying architecture. Studies show that the suggested framework works better in expressions of accuracy, precision, and recall than the standard methods. Worthy results were attained by the OpExBERT model, including 98.55% accuracy, 91.67% precision, 91.25% recall, and 91.14% f-score.*

Keywords *- BERT model, Opinion Extraction (OE), Natural Language Processing (NLP), Machine learning.*

1. Introduction

Applications for NLP are already pervasive and can be found in various forms on several websites and applications. Because transfer learning is included in pre-trained models, NLP use has risen quickly. For problems involving natural language processing, pre-training of the language model has produced outstanding results. Pre-trained models are the finest help for anybody interested in creating or learning an algorithm using an established framework. Due to computational limitations or time constraints, creating a model from scratch is not always possible. For this reason, pre-trained models have become popular. The pre-trained models can be used as a standard against which to compare newly developed models or as a benchmark for enhancing the current model. The feature-based and fine-tuning approaches, both of which are already in existence, can be utilized in order to apply the pre-trained language representations to subsequent tasks. In [1] ELMo, a feature-based technique, the task-specific architectures employed are introduced as additional features to models that have already been pre-trained, which can be found in the Open AI GPT. The Pre-Trained Transformer is a method of fine-tuning that utilizes the downstream tasks for training, all pre-trained parameters for fine-tuning, and very few task-specific parameters. All the aforementioned methods aim to accomplish the same thing: gain extensive linguistic representations by way of pre-training that uses unidirectional language models. [2] In contrast to these limitations, BERT is an architecture that combines a transformer with an encoder stack. The transformer architecture typically uses self-attention on the encoder side and attention on the decoder side in encoder-decoder networks. BERT, specifically, includes an encoder stack that goes beyond the original Transformer design. BERTLARGE has 24 layers compared to BERTBASE's 12 layers. Additionally, both BERTLARGE and BERTBASE have additional attention heads (12 and 16, respectively) and more feedforward networks (768 and 1024 hidden units) equated to the original Transformer architecture. It is worth noting that BERTLARGE has 340 million factors, whereas BERTBASE has 110 million factors. The increased number of layers, attention heads, and hidden units in BERTLARGE and BERTBASE enables these models to capture more complex language dependencies and provide more powerful representations. By incorporating bidirectional context

This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

through the transformer-based encoder stack, BERT addresses the limitations of unidirectional architectures. It allows for more comprehensive language understanding, making it a powerful tool for various natural language processing tasks, including fine-tuning for sentence-level tasks.

This architecture receives a CLS token, formerly a series of words, as its initial input. Initial Input and CLS Token: The input to the BERT model consists of a sequence of tokens. The CLS token (stands for "classification") is pretended to the input sequence as the first token. It serves as a special token that represents the overall sequence for classification tasks.

1.1. Self-Attention and Transformer Layers

The input sequence, including the CLS token, is then processed through multiple transformer layers. Each transformer layer consists of two sub-layers: self-attention and feedforward network.

1.1.1. Self-Attention

Self-attention empowers the model to determine the relative relevance of the various words that make up the input sequence while capturing the connections between those words. It computes attention scores for each word in the sequence, considering all other words in the sequence. This helps the model to apprehend contextual information effectively.

1.2. Feedforward Network

After self-attention, the output is passed through a feedforward neural network. This network applies non-linear transformations to the input representation, further enhancing its expressive power.

1.3. Hidden-Size Vector

As the input passes through each transformer layer, the model generates a hidden-size vector for each token in the sequence. In the case of BERT BASE, the hidden size is typically 768. These hidden vectors capture rich contextual representations of the input words centered on the surrounding context.

1.4. CLS Token Output

The output corresponding to the CLS token can be utilized as a representation of the complete input sequence if the intention is to employ BERT for classification tasks like sentiment analysis. This representation can then be fed into a classifier (e.g., a fully connected layer followed by a softmax activation) for making predictions' models [3-5].

1.5. Need

Opinion extraction helps understand user preferences and sentiments by extracting opinions from user reviews, feedback, and ratings. This information is then used to provide personalized recommendations to users. It is useful in daily life because it may enhance human-computer interaction, government intelligence, commercial intelligence, and citation analysis.

Machine learning can be used to extract opinions from tiny amounts of data, while deep learning performs well with enormous amounts of data. A sizable dataset of 56,000 Amazon smartphone evaluations is employed in this research project to train the machine that will extract opinions. Deep learning is, therefore, appropriate for handling this volume of data.

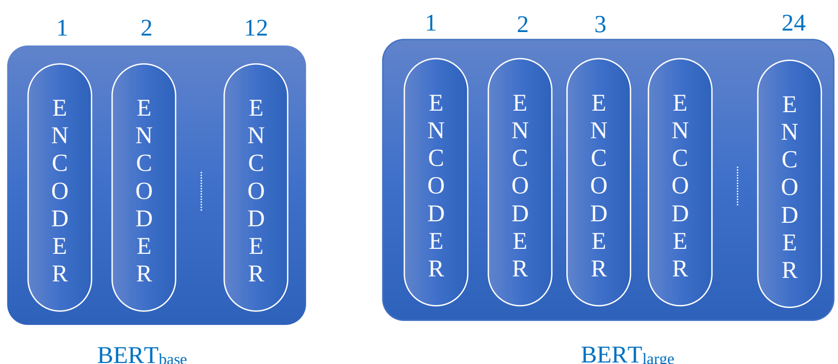

Fig. 1 BERT architecture

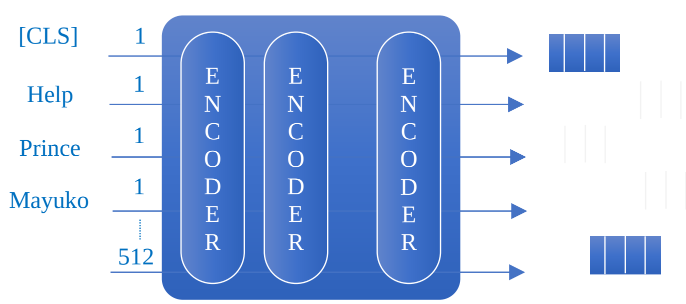

Fig. 2 BERT output as embeddings

1.6. Motivation

E-commerce platforms in the digital era enable millions or billions of people to purchase a wide range of goods. Customers can publicly post their opinions and reviews on particular products on e-commerce platforms. These websites are recognized as excellent informational resources for identifying consumer opinions and predicting product quality based on those opinions. There are a ton of reviews and opinions thanks to the quick development of online platforms. Interpreting the general sentiment of consumers toward a given product is an extremely laborious undertaking. Governments can raise the bar for product quality requirements, firm executives may be able to improve their offerings, and other customers may utilize this information to make judgments. In recent years, online marketplaces have witnessed a growing trend where sellers and merchants encourage customers to provide product feedback and reviews. However, the sheer volume of evaluations can make it challenging for potential buyers to make informed decisions about purchasing a product. This paper proposes a hierarchical attention network-based paradigm to address this issue. The problem at hand is the difficulty faced by potential buyers in analyzing a large number of evaluations to make purchasing decisions. The unstructured nature of textual reviews makes extracting relevant information and identifying key aspects influencing customer opinions challenging.

The suggested solution involves the utilization of a hierarchical attention network-based paradigm, which leverages NLP techniques and BERT embedding. The proposed framework follows a two-step process: preprocessing the Smartphone Review dataset using NLP techniques and extracting word vector representations through BERT embedding. The BERT model is a deep learning model that enables contextual pre-training of unlabeled text from both left and right directions in all layers. This is a departure from previous language representation models that were unidirectional. After the initial pre-training of the BERT model, it becomes possible to fine-tune it with a single output layer for specialized language comprehension tasks, like sentiment analysis and responding to questions. This allows for efficient adaptation without modifying the underlying architecture. The studies conducted on the suggested framework demonstrate its effectiveness in terms of accuracy, precision, and recall when compared to standard methods. The OpExBERT model, a variant of BERT, achieved remarkable results, including 98.55% accuracy, 91.67% precision, 91.25% recall, and 91.14% F-score.

In summary, the proposed hierarchical attention network-based paradigm, incorporating NLP techniques and BERT embedding, presents a solution to the challenge of analyzing vast amounts of evaluations. By leveraging deep learning and contextual pre-training, it offers improved accuracy and effectiveness in understanding customer opinions and aiding potential buyers in making informed purchasing decisions.

2. Literature Survey

This technique [6] aims to enhance the sentiment analysis process by incorporating a multi-polarity orthogonal attention mechanism into a Bi-directional LSTM model. In their study, [6] proposed a novel technique for sentiment analysis. They enhanced the Bi-directional LSTM model with a multi-polarity orthogonal attention-based strategy. By utilizing word embeddings that differ in sentiment polarity, they applied a multi-polarity attention mechanism to capture the characteristics of each sentiment polarity effectively. This approach showed promising results in implicit sentiment analysis. In [7], the author introduced a lexicon-based word polarity recognition approach that leverages customer reviews. Their method establishes semantic relationships between synonym expansion lists of words surrounding the target word and the context expansion list of the target word itself. By incorporating semantic and emotive knowledge, this technique simplifies the comprehension of lengthy reviews, enabling a better understanding of the context. [8] Developed a hybrid recommendation system using a machine learning regression model. This system effectively determines a customer's preferred retailer based on their past purchases without requiring human intervention. The HRS technique achieved a high level of accuracy, as indicated by the mean absolute percentage error number, which approached 98 percent. This approach proves to be efficient and accurate in analyzing consumer sentiment.

The authors in [9] are recommended to conduct sentiment analysis on customer reviews before making marketing decisions. In this study, the researchers examined subjective customer reviews by employing a star rating system and conducting sentiment analysis to categorize the opinions of purchasers. They developed a system incorporating NLP, sentiment analysis, data mining, and clustering techniques to evaluate the sentiment score of specific product characteristics, enabling informed purchasing decisions. The framework determined the product's overall score by considering both the product's pricing and the previously determined sentiment score. Many studies on sentiment analysis for opinion extraction have been conducted [10], and these studies are contrasted in terms of advantages and disadvantages. In a survey of 40 deep learning techniques for sentiment analysis, [11] recommended classifying deep learning models into CNN, RNN, recursive NN, and hybrid approaches. They discuss the benefits and drawbacks of each of the three models. The CNN model [12] has the benefit of quick computing. It is helpful to pull out crucial phrases and terms from documents. Various research [13-18] have been conducted in the past for aspect-level opinion extraction. Authors in [19] developed a Hybrid Recommendation System (HRS) for customer sentiment analysis using a machine learning regression model. A deep CNN-based method for sentiment analysis of Twitter data has been proposed by authors in [20]. The various text processing methods have been compared and examined by authors in [21].

Table 1. Research conducted earlier regarding opinion extraction and sentiment analysis

Study	Proposed Approach	Results/Performance	Benefits	Limitations
Kardakis et al. (2021) [22]	Attention-based model using RNN	Acc = 0.79.% Acc = 0.91% Acc = 0.87%	Methods based on attention have the highest accuracy.	The experiments do not use a variety of dropout rates.
Abdalgader and Al Shibli (2020) [23]	Using WorldNet and SentiWordnet along with contextual information, the word divergence technique.	To determine the precise divergence of the words in relation to the context, record the sentiment score.	Maintain the polarity-based sense or meaning of reviews.	Domain-specific method.
Khalid et al. (2020) [24]	Gradient-boosted support vector machine (GBSVM)	Correctness of Projected (GBSVM) with TF-IDF is 0.93.%	GBSVM's performance is compared to that of four models that are similar to it (GBM, SVM, LR, and RF), and it performs better and provides higher accuracy.	It only applies in particular situations. cases of tf-idf features such as uni, bi and trigram.
Kauffmann et al. (2019) [25]	Techniques in NLP, sentiment analysis, and clustering	Price + Sentiment Score of product to make a decision	Utilizing a strategy will help marketing managers increase the quality of their products.	Features of product extraction are depending upon the designed ontology
Feng et al. (2016) [26]	Deep CNN + word embedding + POS + explicit aspects	P = 0.77% R = 0.72% F1 = 0.75 %	Obtain the sentence's sentiment annotation. improve outcomes	For better performance, more computing power

3. Dataset Description

Information is gathered from Amazon.in and Data is extracted from the "smartphone" result page using web scraping. Web scraping involves storing the necessary URL in a file and subsequently extracting the necessary information from the URL—an approach to organizing the extracted results from opinion extraction.

Saving the results in CSV format allows for easy data storage, manipulation, and analysis. The columns you mentioned—'Mobile name', 'asin number', 'title review', 'user review', and 'star rating score'—are relevant and commonly extracted information from user reviews.

4. Proposed System

NLP pre-training technique BERT was open-sourced. BERT uses attention modeling to comprehend word relationships contextually. BERT's vanilla form has a Transformer with encoders and decoders for reading input text and predicting task outcomes. BERT needs the encoder mechanism to produce language models. BERT model opinion extraction and review classification in natural language processing seems promising. BERT, a strong language model pre-trained on massive volumes of textual material, understands language and context. Using BERT for opinion extraction and classification involves training the model on a labeled dataset of reviews, where each review is labeled with a sentiment score, such as positive, negative, or neutral. The model is then fine-tuned on this dataset, enabling it to classify the latest reviews based on their sentiment. One of the benefits of using BERT for opinion extraction and classification is that it can handle long sequences of text, which is often the case with reviews. Also, BERT can capture contextual information, which is important for understanding the sentiment of a review. Using BERT for opinion extraction and classification techniques improves the accuracy of sentiment analysis and provides valuable insights for businesses looking to understand customer feedback.

The proposed hierarchical attention network-based paradigm addresses the challenge of analyzing and making informed decisions based on vast amounts of product evaluations in online marketplaces. The framework utilizes NLP techniques and BERT embedding to preprocess the Smartphone Review dataset and extract word vector representations that capture word context.

BERT is a deep learning model designed for pre-training unlabeled text by considering both left and right contextual information at all layers. This differs from previous language representation models that were unidirectional. By pre-training BERT and fine-tuning it with just one output layer, it becomes adaptable to various language understanding tasks, including opinion mining and question answering, without requiring significant modifications to its underlying architecture.

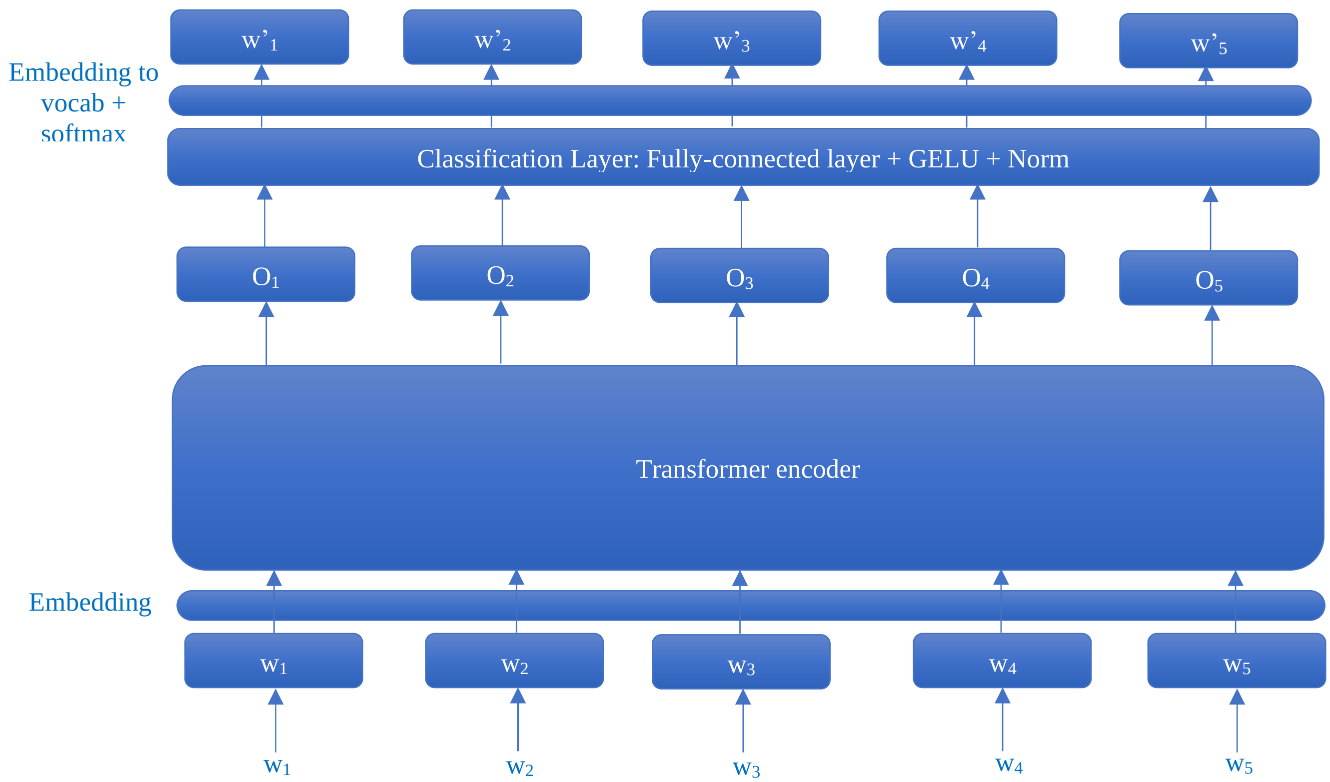

Fig. 3 BERT proposed diagram

4.1. OpExBERT

The basic notion of Transformer, which BERT relies on totally, is the attention mechanism, which aids in understanding the contextual relationship between the words of the phrase. As previously said, an encoder is the sole component necessary for BERT operation. An encoder receives a string of tokens as input, which are then converted into vectors before being passed to the network for further processing. In order to carry out this processing, the inputs must first be embellished with additional metadata in the following manner:

4.2. Token Embeddings

Tokens [CLS] at the beginning and [SEP] at the end detect sentences.

4.3. Segment Embeddings

Markers like Sentence A or Sentence B help encoders distinguish between sentences.

4.3.1. Positional Embeddings

Positional embedding adds a token to a sentence to indicate word order. A proposed system for opinion extraction and classification of reviews using the BERT model would typically involve the following steps:

Data Collection

Collect a large dataset of reviews, such as from e-commerce websites, social media platforms, or review websites.

Data Preprocessing

Preprocess the data by cleaning the text, removing stopwords, and performing other necessary preprocessing steps.

Labeling the Data

Label each review with a sentiment score, such as positive, negative, or neutral. Train the BERT model: Fine-tune the pre-trained BERT model on the labeled dataset, using techniques such as transfer learning to adapt the model to the specific domain and task at hand.

Evaluation

Measure the model's performance using metrics like accuracy, precision, recall, and F1-score on a validation dataset.

Deployment

Deploy the trained model to classify new reviews and extract opinions automatically.

Interpretation

Interpret the results and gain insights into customer sentiment, identify areas for improvement, and take appropriate actions to enhance the customer experience.

Opinion Extraction using BERT can be formulated as a sequence labeling task. Given a sequence of input tokens, the task is to predict the opinion label for each token in the sequence.

Let us define:
N: Number of tokens in the input sequence
V = {v_1, v_2, ..., v_N}: Input tokens
U = {u_1, u_2, ..., u_N}: Predicted opinion labels
The goal is to find the optimal predicted opinion labels, Y*, that maximizes the probability of the sequence given the input tokens:

$$Y^* = \text{argmax}(P(U|V)) \quad (1)$$

4.4. OpExBERT Algorithm

Step 1: Input Preparation
tokens = tokenizer.tokenize (input_text)
tokens = ['[CLS]'] + tokens + ['[SEP]']
input_ids = tokenizer.convert_tokens_to_ids(tokens)
input_mask = [1] * len(input_ids)
segment_ids = [0] * len(input_ids)

Padding
padding_length = max_seq_length - len(input_ids)
input_ids += [0] * padding_length
input_mask += [0] * padding_length
segment_ids += [0] * padding_length

Step 2: Model Prediction
logits, hidden_states = BERT_model(input_ids, input_mask, segment_ids)

Step 3: Opinion Labeling
opinion_labels = [ ]
for i in range(1, len(tokens)-1):
 token_logits = logits[i] # Logits for the current token
 predicted_label = argmax(token_logits) # Predicted opinion label for the token   opinion_labels.append(predicted_label)

Step 4: Output Opinion Extraction
opinion_spans = extract_opinion_spans(opinion_labels, tokens)

Output the extracted opinion spans or use them for further processing. In the pseudocode above, extract_opinion_spans is a function that takes the predicted opinion labels and the input tokens as input and identifies contiguous spans of tokens representing opinions.

5. Experiments Results

Python is used to implement the suggested model, and the high-level neural network API Keras is utilized for the model's deep learning functionalities. Opinion extraction and classification of reviews using the BERT model- The Amazon Smartphone review database, which comes from pages on amazon.in, is used for the BERT model. 56000 smartphone comments are gathered after preprocessing from 250000 reviews scraped from the browser. From the 56000 reviews, 80%, or 44,800 reviews, are specified for training, 10%, or 5600 reviews, are used for validation, and the remaining 10%, or 5600 reviews, are used for testing. Experiments allow for a maximum of 250000-word features, 50 sentences, 40 words, and an embedding dimension size of 100.

5.1. Evaluation Metrics

The OpExBERT model's performance is evaluated using the accuracy, precision, recall, and F-Score. The percentage of accurately predicted groups of opinions out of all the reference groups of opinions is how recall is represented. The proportion of groups of opinions that were accurately predicted to all of those that were successfully predicted can be used to measure precision. Accuracy, precision, and recall are commonly used performance metrics for evaluating opinion extraction models. These metrics are often computed based on the number of predictions generated by the model that were true positive (TP), true negative (TN), false positive (FP), and false negative (FN).

5.1.1. Accuracy

Accuracy is a metric that measures the proportion of correct predictions made by the model's overall predictions. It is defined as:

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN) \quad (2)$$

5.1.2. Precision

Precision is a metric showing how many of the model's correct positive predictions out of all are right. It is defined as:

$$\text{Precision} = TP / (TP + FP) \quad (3)$$

5.1.3. Recall

Recall is the model's positive prediction accuracy compared to the dataset's actual positive cases. It is defined as: [21]

$$\text{Recall} = TP / (TP + FN) \quad (4)$$

A model's performance, as restrained in terms of accuracy, precision, F-score, and recall, is presented in Table 2. This performance was trained using a variety of batch sizes. According to the chart, we can deduce that the size of the batches used in the model has a considerable bearing on its overall performance. The lowest level of accuracy is achieved with a batch size of 5, whereas bigger batch sizes of 12 and 15 achieve greater levels of accuracy, precision, and F-score, respectively. On the other hand, a batch size of 15 comes out as the best performer, accomplishing high levels of accuracy, precision, F-score, and recall. In addition, it is important to observe that a batch size of 32 results in the lowest recall, although achieving an exceptionally good accuracy and F-score.

Table 2. Comparison of different batch size

Batch sizes	Accuracy	Precisions	F_Scores	Recall
5	28.75	90.46	90.35	90.31
8	32.89	84.46	83.79	78.73
12	36.53	90.85	90.43	76.42
15	98.55	91.67	91.14	91.25
20	78.61	84.87	84.33	85.00
23	84.32	78.70	78.73	79.06
32	73.52	77.23	77.65	90.87

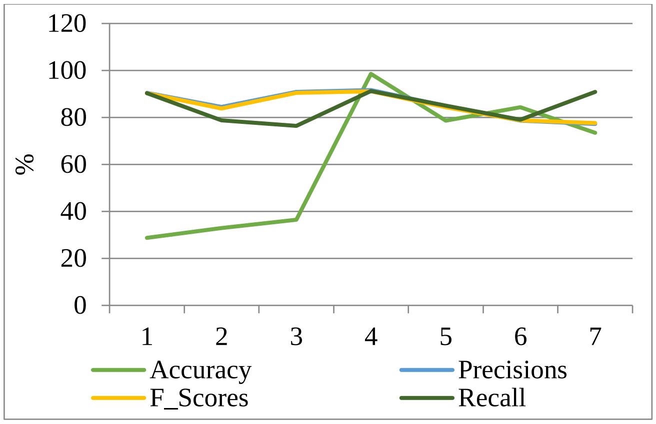

Fig. 4 Comparison of different batch sizes

Table 3. Performance comparison of opinion extraction in comparison to existing methods

Work	Accuracy	Precision	F_Score	Recall
LSTM+ Glove	73.06	80.59	80.87	81.16
GRU+ Glove	56.04	68.00	68.53	69.06
LSTM+ Word2Vec	85.00	74.00	76.41	79.00
GRU+ SVM	75.82	59.00	55.34	52.12
OpExBERT (proposed)	**98.55**	**91.67**	**91.14**	**91.25**

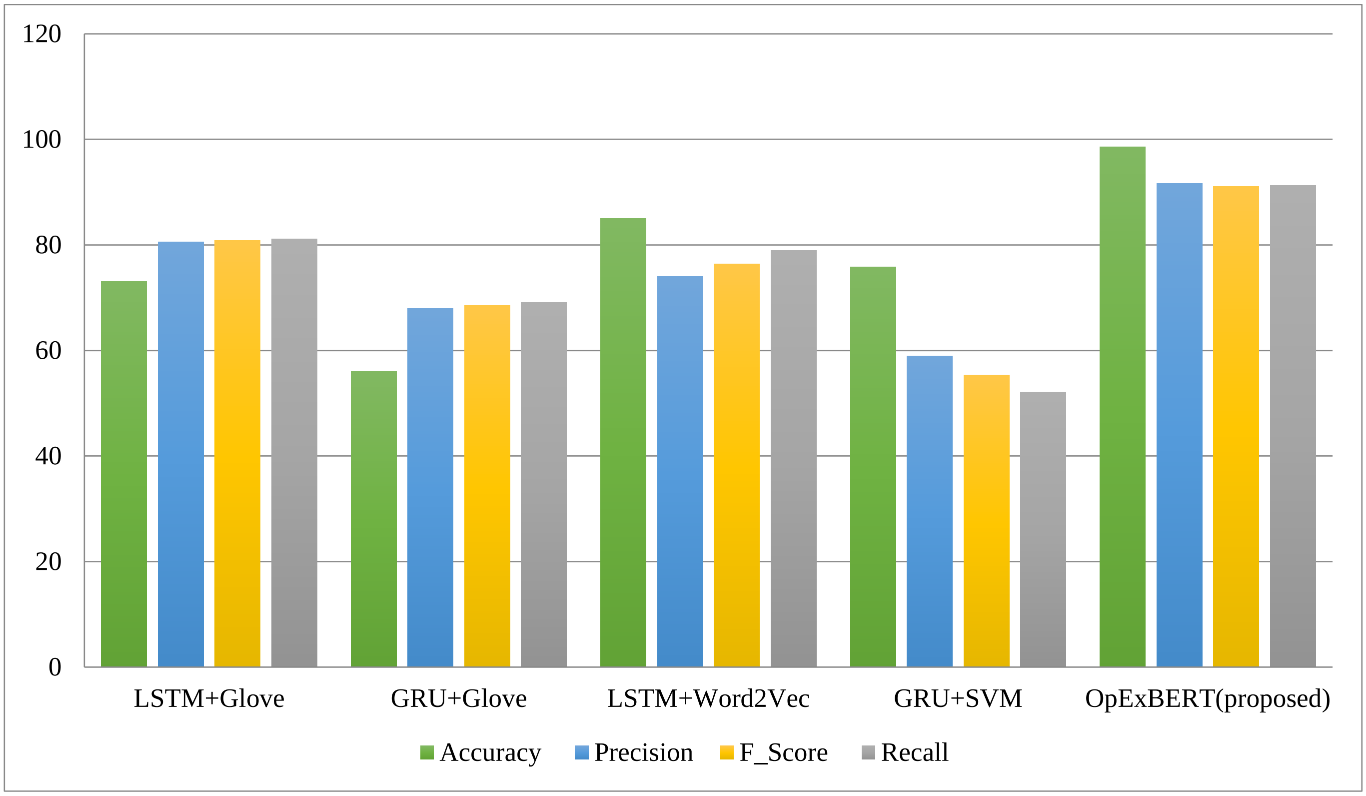

Fig. 5 Performance comparison of opinion extraction in comparison to existing methods

Table 3 shows the performance of different models for sentiment analysis, measured in terms of accuracy, precision, F-score, and recall. The models compared are:

LSTM+Glove: A model using LSTM networks and pre-trained GloVe embeddings.

GRU+Glove: A model using Gated Recurrent Unit (GRU) networks and pre-trained GloVe embeddings.

LSTM+Word2Vec: A model using LSTM networks and pre-trained Word2Vec embeddings.

GRU+SVM: A model using GRU networks and a Support Vector Machine (SVM) classifier.

OpExBERT (proposed): A BERT model for opinion extraction and classification.

Based on Table 3, we can see that the OpExBERT model outdoes all former models, achieving the highest accuracy, precision, F-score, and recall. This suggests that using BERT for opinion extraction and classification is a powerful technique that can significantly improve the performance of sentiment analysis models. Among the other models, the LSTM+Word2Vec model performs relatively well, achieving high accuracy and recall. However, it has lower precision and F-score compared to OpExBERT. The other models, including LSTM+Glove, GRU+Glove, and GRU+SVM, achieve lower overall performance, with accuracy ranging from 56.04% to 75.82%.

Epoch 1/15
160/160 [==============================] - 166s 965ms/step - loss: 1.0982 - accuracy: 0.5719 - val_loss: 0.8515 - val_accuracy: 0.6641

Epoch 2/15
160/160 [==============================] - 153s 959ms/step - loss: 0.6720 - accuracy: 0.7477 - val_loss: 0.5335 - val_accuracy: 0.8125
.
.
.
-
-
-
-

Epoch 14/15
160/160 [==============================] - 154s 960ms/step - loss: 0.0303 - accuracy: 0.9855 - val_loss: 0.3914 - val_accuracy: 0.9219

Epoch 15/15
160/160 [==============================] - 154s 960ms/step - loss: 0.0280 - accuracy: 0.9855 - val_loss: 0.3957 - val_accuracy: 0.9281

Table 4. Comparison result with existing work

Work	Models	Accuracy (%)
Ratmele et.al.(2022) [15]	OpExHAN mode	94.68
Kardakis et al. (2021) [22]	Attention-based model using RNN	79.97
Khalid et al. (2020) [24]	GBSVM	93.0
Ma et al. (2019) [17]	Feature-based integrating memory networks (RNN)	82.03
OpExBERT (Proposed)	**OpExBERT**	**98.55**

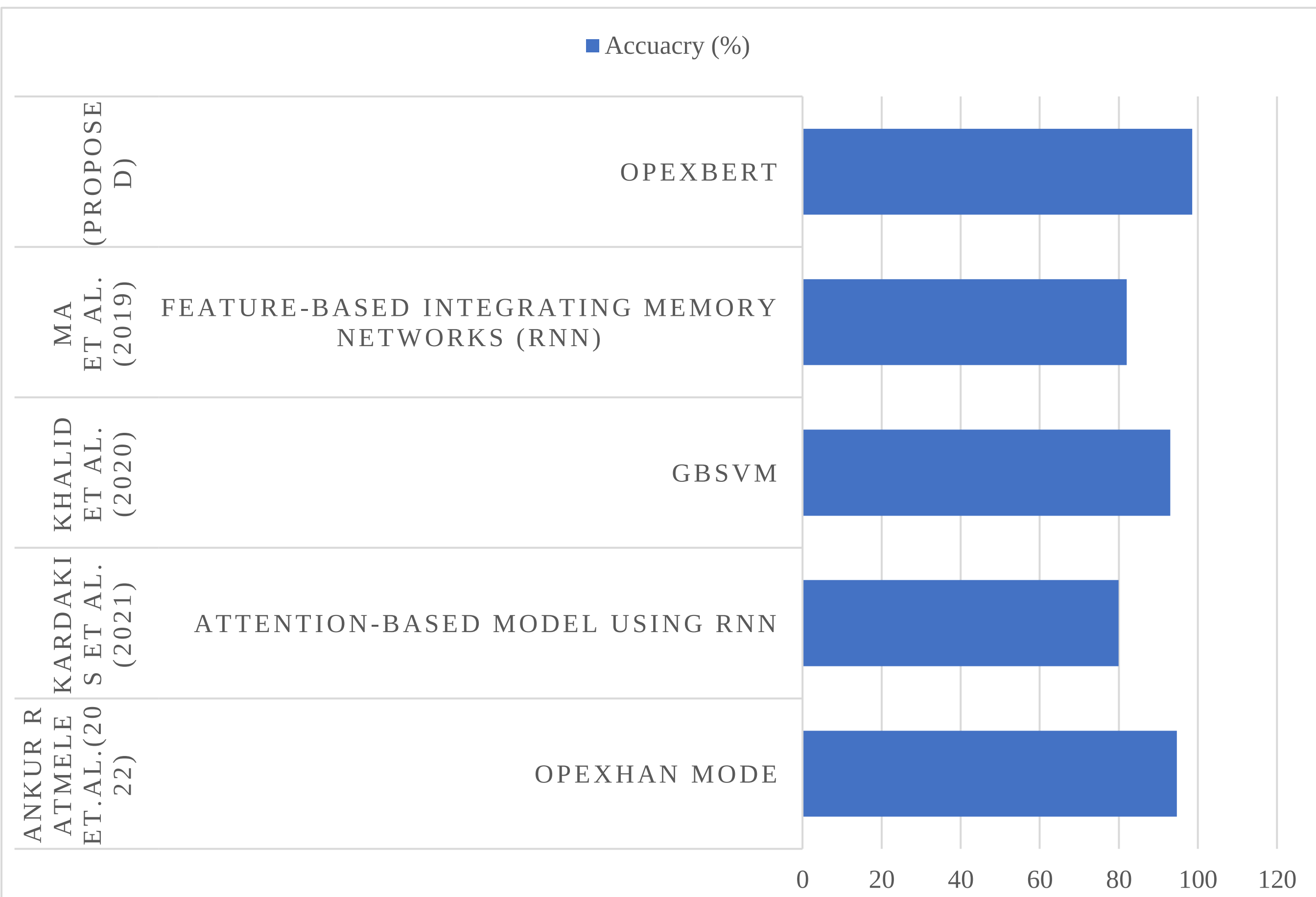

Fig. 6 Comparison results with existing work

The output shows the training progress of a deep-learning model for 15 epochs. The training process involved 160 iterations. The output shows that the model's loss decreases over the course of the training process. At the end of the final epoch, the loss is 0.0280. This indicates that the model is improving and getting closer to the optimal set of weights to minimize the error between its predicted and actual outputs.

Additionally, the output shows that the model's accuracy also improves over the course of training, reaching 0.9855 at the end of the final epoch. This indicates that the model is becoming more accurate in its predictions as it learns from the training data.

Additionally, validation measures are included in the output, which gauge how well the model performs on a unique data set. The validation accuracy is 0.9281, which is lower than the training accuracy. This suggests that the model may be overfitting to the training data and may not generalize well to new data.

6. Conclusion

In this research, the BERT technique is used to extract opinions from a large amount of the reviews of Smartphones. The BERT model uses a bidirectional training strategy of a popular attention model known as the Transformer language modeling. This approach enabled a greater comprehension of the context of the language and word embedding and did exceptionally well on state-of-the-art methodologies. Using Amazon's dataset, many different tests are done to evaluate the performance of the proposed approach with that of numerous baseline existing methods.

As a result, the presented method's viability is shown through various evaluation criteria. The proposed model has a precision of approximately 91.61 and 91.25%, a recall of approximately 91.25%, and an accuracy of 98.55%, and an f-score of 91.14, all of which are admirable statistics. In upcoming research, the focus will be on extracting aspect-based opinions to make a purchasing choice based on a prediction of the customer's view of the product's features.

References

[1] A. Palaniammal, and P. Anandababu, "Robust Sarcasm Detection Using Artificial Rabbits Optimizer with Multilayer Convolutional Encoder-Decoder Neural Network on Social Media," *SSRG International Journal of Electronics and Communication Engineering*, vol. 10, no. 5, pp. 1-13, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[2] Jinzhan Feng, Shuqin Cai, and Xiaomeng Ma, "Enhanced Sentiment Labeling and Implicit Aspect Identifcation by Integration of Deep Convolution Neural Network and Sequential Algorithm," *Cluster Computing*, vol. 22, pp. 5839-5857, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[3] Himanshu Thakur, and Aman Kumar Sharma, "Supervised Machine Learning Classifiers: Computation of Best Result of Classification Accuracy," *International Journal of Computer Trends and Technology*, vol. 68, no. 10, pp. 1-8, 2020. [CrossRef][Google Scholar] [Publisher Link]

[4] Wanliang Tan, Xinyu Wang, and Xinyu Xu, "Sentiment Analysis for Amazon Reviews," *International Conference*, pp. 1-5, 2019. [Google Scholar] [Publisher Link]

[5] Anh-Dung Vo, Quang-Phuoc Nguyen, and Cheol-Young Ock, "Opinion-Aspect Relations in Cognizing Customer Feelings via Reviews," *IEEE Access*, vol. 6, pp. 5415-5426, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[6] Yequan Wang et al., "Attention-Based LSTM for Aspect-Level Sentiment Classifcation, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 606-615, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[7] Rui Wang et al., "A Survey on Opinion Mining: from Stance to Product Aspect," *IEEE Access*, vol. 7, pp. 41101-41124, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[8] Jiyao Wei et al., "BiLSTM with Multipolarity Orthogonal Attention for Implicit Sentiment Analysis," *Neurocomputing*, vol. 383, pp. 165-173, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[9] Chuhan Wu et al., "A Hybrid Unsupervised Method for Aspect Term and Opinion Target Extraction," *Knowledge-Based Systems*, vol. 148, pp. 66-73, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[10] Zichao Yang et al., "Hierarchical Attention Networks for Document Classification," *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1480-1489, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[11] Shanshan Yi, and Xiaofang Liu, "Machine Learning Based Customer Sentiment Analysis for Recommending Shoppers, Shops Based On Customers' Review," *Complex and Intelligent Systems*, vol. 6, pp. 621-634, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[12] Min Zhang et al., "Mining Product Innovation Ideas from Online Reviews," *Information Processing and Management*, vol. 58, no. 1, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[13] Petr Hajek, Aliaksandr Barushka, and Michal Munk, "Opinion Mining of Consumer Reviews Using Deep Neural Networks with Word-Sentiment Associations," *IFIP International Conference on Artificial Intelligence Applications and Innovations*, pp. 419-429, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[14] Tanjim Ul Haque, Nudrat Nawal Saber, and Faisal Muhammad Shah, “Sentiment Analysis on Large Scale Amazon Product Reviews,” *2018 IEEE International Conference on Innovative Research and Development (ICIRD)*, pp. 1-6, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[15] Ankur Ratmele, and Ramesh, “Thakur OpExHAN: Opinion Extraction Using Hierarchical Attention Network from Unstructured Reviews,” *Social Network Analysis and Mining*, vol. 12, no. 148, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[16] Doaa Mohey El-Din Mohamed Hussein, “A Survey on Sentiment Analysis Challenges,” *Journal of King Saud University - Engineering Sciences*, vol. 30, no. 4, pp. 330-338, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[17] Ruixin Ma et al., “Featurebased Compositing Memory Networks for Aspect-Based Sentiment Classification in Social Internet of Things,” *Future Generation Computer Systems*, vol. 92, pp. 879-888, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[18] Yi Tay, Luu Anh Tuan, and Siu Cheung Hui, “Learning to Attend via Word-Aspect Associative Fusion for Aspect-Based Sentiment Analysis,” *Proceedings of the AAAI Conference on Artifcial Intelligence*, vol. 32, no. 1, pp. 1-8, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[19] Peter D. Turney, “Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classifcation of Reviews,” *arXiv*, pp. 1-8, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[20] Prajwal Madhusudhana Reddy, “Conducting Sentiment Analysis on Twitter Tweets to Predict the Outcomes of the Upcoming Karnataka State Elections,” *SSRG International Journal of Computer Science and Engineering* , vol. 10, no. 6, pp. 22-35, 2023. [CrossRef] [Publisher Link]

[21] Zhao Jianqiang, Gui Xiaolin, and Zhang Xuejun, “Deep Convolution Neural Networks for Twitter Sentiment Analysis,” *IEEE Access*, vol. 6, pp. 23253-23260, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[22] Z. Jianqiang, and G. Xiaolin, “Comparison Research on Text Preprocessing Methods on Twitter Sentiment Analysis,” *IEEE Access*, vol. 5, pp. 2870-2879, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[23] Spyridon Kardakis et al., “Examining Attention Mechanisms in Deep Learning Models for Sentiment Analysis,” *Applied Sciences*, vol. 11, no. 9, pp. 1-14, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[24] Khaled Abdalgader, and Aysha Al Shibli, “Experimental Results on Customer Reviews Using Lexicon-Based Word Polarity Identifcation Method,” *IEEE Access*, vol. 8, pp. 179955-179969, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[25] Madiha Khalid et al., “GBSVM: Sentiment Classification from Unstructured Reviews Using Ensemble Classifier,” *Applied Sciences*, vol. 10, no. 8, pp. 1-20, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[26] Erick Kauffmann et al., “Managing Marketing Decision-Making with Sentiment Analysis: An Evaluation of the Main Product Features Using Text Data Mining,” *Sustainability*, vol. 11, no. 15, pp. 1-19, 2019. [CrossRef] [Google Scholar] [Publisher Link]