ARTICLE

DeBERTa-GRU: Sentiment Analysis for Large Language Model

Adel Assiri[1], Abdu Gumaei[2,*], Faisal Mehmood[3,*], Touqeer Abbas[4] and Sami Ullah[5]

[1]Department of Informatics for Business, College of Business, King Khalid University, Abha, 61421, Saudi Arabia

[2]Department of Computer Science, College of Computer Engineering and Sciences, Prince Sattam bin Abdulaziz University, Al-Kharj, 11942, Saudi Arabia

[3]School of Electrical and Information Engineering, Zhengzhou University, Zhengzhou, 450001, China

[4]Department of Computer Science and Technology, Beijing University of Chemical Technology, Beijing, 100029, China

[5]Department of Computer Science, Government College University Faisalabad, Faisalabad, Punjab, 38000, Pakistan

*Corresponding Authors: Abdu Gumaei. Email: a.gumaei@psau.edu.sa; Faisal Mehmood. Email: faisalmehmood685@uaf.edu.pk

ABSTRACT

Modern technological advancements have made social media an essential component of daily life. Social media allow individuals to share thoughts, emotions, and ideas. Sentiment analysis plays the function of evaluating whether the sentiment of the text is positive, negative, neutral, or any other personal emotion to understand the sentiment context of the text. Sentiment analysis is essential in business and society because it impacts strategic decision-making. Sentiment analysis involves challenges due to lexical variation, an unlabeled dataset, and text distance correlations. The execution time increases due to the sequential processing of the sequence models. However, the calculation times for the Transformer models are reduced because of the parallel processing. This study uses a hybrid deep learning strategy to combine the strengths of the Transformer and Sequence models while ignoring their limitations. In particular, the proposed model integrates the Decoding-enhanced with Bidirectional Encoder Representations from Transformers (BERT) attention (DeBERTa) and the Gated Recurrent Unit (GRU) for sentiment analysis. Using the Decoding-enhanced BERT technique, the words are mapped into a compact, semantic word embedding space, and the Gated Recurrent Unit model can capture the distance contextual semantics correctly. The proposed hybrid model achieves F1-scores of 97% on the Twitter Large Language Model (LLM) dataset, which is much higher than the performance of new techniques.

1 Introduction

Sentiment analysis is a procedure that examines how people's attitudes, sentiments, and feelings are expressed in their written words. Due to its significance, sentiment analysis is one of the most well-known Natural language Processing (NLP) and has become a popular research topic in recent years [1,2]. Several Machine Learning (ML) and deep learning methods have been suggested for the

problem of relation classification, utilizing sentiment analysis such as Robustly Optimized Bidirectional Encoder Representations from Transformers attention (RoBERTa). Long Short-Term Memory (LSTM) [3], Improved BERT [4], BERT and Deep Convolutional Neural Network (CNN) (BERT-DCNN) [5]. Erfan et al. [6] propose a technique for adaptable aspect-based lexicons in sentiment categorization. A dynamic lexicon is automatically updated, allowing for more precise grading of context-related ideas [7]. Applications of multimodal sentiment analysis, along with its opportunities, difficulties, and related fields [8]. The study conducted by [9] focused on using sentiment analysis to develop an intelligent society that relies on public services.

These days, both social media and business are affected by sentiment analysis. Thanks to the social media industry's rapid expansion, everyone now has access to the internet to express their thoughts and opinions. Therefore, sentiment analysis is crucial for determining what customers or reviewers think. In addition, sentiment analysis is essential for examining how the general public responds to social media issues. The opinions of the public influence social media decisions.

Recently, we have seen a significant rise in interest in large language models (LLMs) in both academic and industrial fields [10–12]. Existing research [13] has shown that LLMs function exceptionally well, raising the possibility that they could become AGI in this period. Unlike earlier models that were limited to tackling particular tasks, LLMs can solve various problems. People with essential information needs increasingly utilize LLMs due to their excellent performance in handling a variety of applications, such as generic natural language tasks and specific domains [14–16].

Starting with ChatGPT 3 in November 2022, Microsoft's Bing Chat and Google Bard have quickly extended these capabilities with more LLMs that add links to search results and other material [17]. Hence, ChatGPT has been improved to include creating images and interacting with different data types, a function that has also been added by Google and Microsoft [18]. Thus, we now have technologies that can make human-sounding text, but not necessarily accurately. Several recent articles have highlighted how these tools can pass the bar test or successfully finish academic tasks. The capacity of this technology to iteratively refine subsequent outputs based on earlier reactions and outputs is a significant advancement. ChatGPT is available in two versions: A free version that uses the model version 3.5 and a commercial version that, at the time of writing, employs the model version 4.0. It is possible that OpenAI included changes to the commercial version due to user input from the free version. Version 4.0 also supports image inputs, which is a significant feature. As an illustration, it could generate the necessary code from a drawing of a website design. There are additional platforms available, such as the Bard platform from Google, and additional LLMs, such as the open-access, multilingual BLOOM LLM with 176 B parameters [19], which can be used to power chat-like interfaces and to build other applications. The capacity to properly induce desired outputs is not always straightforward because of the iterative nature of GPT-3.5, GPT-4, BARD, Claude, and cohere and the sensitivity to the exact choice of the text prompt. Because of this, "prompt engineering" and being a "prompt engineer" are significant.

The complexity of the lexicon and text distance relationships make sentiment analysis difficult. For sentiment analysis, various ML techniques were put forward, particularly sequence models that store the distance relationships of the text. So, we have proposed a model to comprehensively evaluate and compare various LLMs across a diverse range of natural language understanding and generation tasks.

- A hybrid deep learning model that combines the DeBERTa and GRU models is proposed for sentiment analysis.
- The GRU model encodes the long-distance temporal relationships in the word embedding.

- The DeBERTa model aims to generate word or subword tokenization and word embedding.
- More lexically rich training samples improve the model's generalization ability, and the imbalanced dataset problem is solved.
- The empirical results demonstrate that the proposed DeBERTa-GRU method performs significantly better than existing methods.

There are five main parts of this study. In the first section, we provide some context for the research and briefly describe the unique contributions of this study. Section 2 outlines the current sentiment analysis techniques. Section 3 examines the proposed DeBERTa-GRU's inner workings, architecture, and other technical specifications. In Section 4, the results of the experiments are shown, along with an analysis of how the suggested method compares to other methods already in use. The conclusion is described in Section 5 at the end of the paper.

2 Literature Review

This part will discuss the most recent developments in sentiment analysis, focusing on modern methods classified as ML and deep learning approaches.

2.1 Machine Learning

Three ML techniques—Naive Bayes, Support Vector Machine (SVM), and K-Nearest Neighbors (KNN) were compared [20,21]. The data collection about the Republic of Indonesia's 2019 presidential contenders was scraped from Twitter. Positive and negative classifications of data samples were assigned to the models. Before the commencement of the training phase, several preprocessing techniques were employed, including text mining, tokenization, and text parsing. The training phase utilized 80% of the dataset, while 20% was allocated for testing. The research demonstrates that the Naive Bayes approach, with a rate of 75.58%, was the most accurate, followed by the KNN method, at 73.34%, and the SVM method, at 63.99% [20].

Multinomial Naive Bayes, Support Vector Classifier (SVC), and linear kernels were also evaluated for their efficacy in sentiment analysis [22]. The Twitter dataset contains airline reviews. The dataset includes 10 K positive, negative, and neutral tweets. The data was first cleaned using preprocessing techniques like stemming, removing URLs, and removing stop words. In the analysis, 33% of the data is used for evaluation, while the remaining 67% is put to good use in the form of training. SVC attained 82.48% accuracy in the experiments, while Multinomial Naive Bayes achieved 76.56% accuracy [22].

Madhuri [23] compared four ML methods for sentiment analysis. The Twitter-compiled Indian Railways Case Study dataset. Positive, harmful, or neutral labels were on data samples. According to the experimental findings, C4.5, Naive Bayes, SVM, and Random Forest obtained accuracy levels of 89.5%, 89%, 91.5%, and 90.5%, respectively [23].

Prabhakar et al. [24] developed an AdaBoost model for sentiment analysis of Twitter data from US airlines. Data preparation was done to remove extraneous data from the text. Additionally, several data mining techniques were used to comprehend the relationships between the dataset's components. Using those terms, the top 10 US Airlines were referenced in the dataset obtained via Skytrax and Twitter. In the experiments, 25% of the data were set aside for testing and 75% for training [24]. To get the greatest F1-score of 68%, six ML models were used to assess US Airways tweets for sentiment using AdaBoost's boosting and bagging approaches [25]. Preprocessing procedures were performed, such as stop word removal, punctuation removal, case folding, and stemming. The Bag of Words technique was employed for feature extraction. The dataset utilized in this study was obtained from

two reputable sources, namely CrowdFlower and Kaggle. These sources provided data about six major airlines operating inside the United States. The dataset comprises a total of 14,640 samples, which are classified into three distinct classes: Positive, negative, and neutral. Training and testing containing 70% and 30% of data were created from the dataset. In that sequence, Naive Bayes, Extreme Gradient Boosting (XgBoost), SVM, and Logistic Regression all achieve accuracy levels of 83.31%, 81.81%, 78.55%, and 75.33%, respectively [25].

2.2 Deep Learning

Younas et al. [26] developed two methods based on deep learning for multilingual social media sentiment analysis. The dataset came from Twitter during the 2018 Pakistan general election. There are a total of 20,375 tweets in the collection. They are written in two different languages: English and Roman Urdu. The dataset was divided into positive, negative, and neutral categories. They took (80:20)% of the data for training and testing. The authors analyzed the effectiveness of BERT models, namely Multilingual BERT (mBERT) and Cross-lingual Language Model-DeBERTa (XLM-DeBERTa) [26].

Sentiment analysis in self-collected Tamil tweets was performed using a character-based Deep Bidirectional LSTM (DBLSTM) technique [27]. The dataset includes 1,500 tweets that can be categorized as either positive, negative, or neutral. As a first step, they performed data preprocessing to clean the text of any extraneous characters. The Word2Vec pre-trained model-based DBLSTM word embedding was then used to represent the cleaned data. They used 80% and 20% of datasets for training and testing, respectively. DBLSTM archived with 86.2% accuracy [27].

According to Thinh et al. [28], using 1D-CNN and Recurrent Neural Network (RNN) is proposed as a viable approach for conducting sentiment analysis. The data came from the IMDb dataset, which has 50,000 reviews of good and bad movies. They used (50:50)% data for testing and training. The 1-dimensional CNN (1D-CNN) architecture incorporates convolutional layers with 128 and 256 filters. The RNN layers consist of a total of 128 units, which include LSTM, Bi-LSTM, and GRU. According to the experimental findings, the 1D-CNN with the GRU model produced a maximum accuracy of 90.02% [28].

In the same way, Kumar et al. [29] suggested a Sentiment Analysis Bidirectional-LSTM (SAB-LSTM). The model comprises 196 Bi-LSTM units, 128 Embedding layers, four thick layers, and a SoftMax activation function in the classification layer. The study said that adding more layers could help avoid the problem of overfitting and dynamically optimize the model parameters. The data collection includes 80,689 samples from five different sentiment classifications obtained from social media users like Twitter, YouTube, and Facebook, as well as from news stories. The training dataset comprised 90% of the whole dataset, while the remaining 10% was allocated for testing purposes. The outcomes of the studies showed that the Sentiment Analysis Bidirectional-LSTM (SAB-LSTM) model reached the standard LSTM models [29].

Dhola et al. [30] analyzed how well different sentiment analysis methods performed, including the SVM, Multinomial Naive Bayes, LSTM, and BERT. Tokenization, stemming, lemmatization, stop word and punctuation removal, and similar preparation techniques were used. The dataset includes 1.6 million tweets that have been classified as either positive or negative. Training and evaluation data comprised 80% and 20% of the dataset. Based on the results, BERT was shown to be the most effective method (85.4% accurate) [30]. To address the sentiment analysis of 5,000 Tweets written in Bangla, the authors of [31–33] used LSTM. By removing whitespace and punctuation, the dataset was cleaned. The dataset was divided into three parts: Training, validation, and testing in the experiments. The best

accuracy (86.3% after hyperparameter adjustment) was achieved with an architecture consisting of 5 LSTM layers, each of size 128, with a batch size of 25 and a learning rate of 0.0001 [31].

Table 1 summarizes the relevant works, detailing the methodologies and datasets employed. There is still much to learn about the Transformers models. RNN and transformer models each have their benefits. RNN effectively encodes long-range dependencies, and Transformers expedites computation via parallel processing. Therefore, this research examines the combination of Transformers and RNN.

Table 1: Summary of related work

Methods	Datasets	Accuracy (%)
LSTM-Bi-LSTM [34]	Saudi tweets	94
NBC-LS [35]	Sentiment140	90
LSTM [31]	Bangla tweets	86.3
BLSTM [27]	Self-collected Tamil tweets	86.2
ID-CNN [28]	IMDb	90.02
SAB-LSTM [29]	Social media	90
SVM, DT, RF [24]	Tweeter US airline	74,72,71
NB, SVM and KNN [20]	Indonesia tweets	75.58, 63.99, 73.34
SVC, MNB [22]	Airline reviews	82.48, 76.56

Many studies have used neural networks to learn word embeddings since deep learning and neural networks became well-known [36]. Yoshua et al. [37] have provided a groundbreaking effort in this area. They describe a neural probabilistic language model that learns the probability function for word sequences based on these word representations and a continuous representation for words simultaneously.

Some recent efforts have attempted to learn sentiment-tailored word embeddings by storing the sentiment polarity of texts. Maas et al. [38] presented a probabilistic topic model that infers the polarity of a phrase based on the embeddings of each word it contains. Igor et al. [39] used logistic regression to re-embed existing word embeddings by looking at sentence mood supervision as a regularization item. Duyu et al. [40] used sentiment embeddings to build sentiment lexicons, classify at the phrase level and analyze sentiment at the word level.

3 Method and Material

The steps for the proposed DeBERTa-GRU for sentiment analysis are described in Section 3.3. First, preparation operations are carried out on the corpus to eliminate extraneous tokens or symbols. By using Lexicon-Based Approaches, we labelled the dataset. Finally, the DeBERTa-GRU model trains and classifies the balanced dataset.

The proposed model, known as the DeBERTa-GRU model, is a combination of the Decoding-enhanced BERT with disentangled attention (DeBERTa) method [41] and the Gated Recurrent Unit (GRU) [42]. To efficiently translate the tokens into meaningful embedding space, the proposed model uses the pre-trained DeBERTa weights. The GRU is then fed the resulting word embeddings to extract the most important semantic information.

3.1 Data Collection

Data collection is a method for transforming unprocessed data into a practical and usable format. Twitter data is gathered via web scraping methods and a third-party program named Snscrape. Snscrape is a trustworthy tool for getting data like user profiles, hashtags, or pertinent search results, even though it is not a part of the official Twitter Application Programming Interface (API). Snscrape is a scraper for social network services (SNS) that can extract pertinent postings associated with the study's goals. Fig. 1 shows the phases of data gathering.
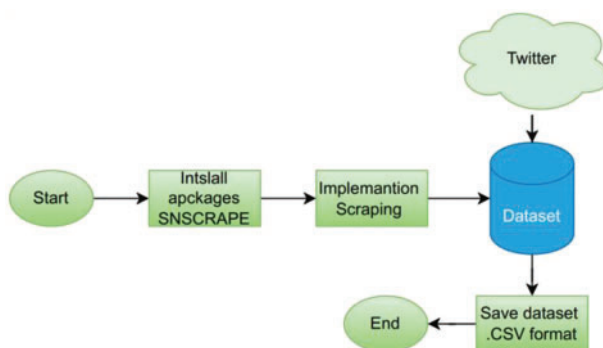
Figure 1: Data collection stages

The Twitter LLM Sentiment dataset was collected from 01 March 2022 to 31 December 2023. The dataset contains 9640 tweets. The hashtag identifies the LLM models used in the dataset, which include GPT-3.5, GPT-4, BARD, Claude, and Cohere. The collected data will be used for sentiment analysis. The frequency of LLM models is shown in Fig. 2. The statistics of collected data are given in Table 2.

Table 2: Statistics on the tweet's dataset

Metadata	Positive	Negative	Neutral
Total number of tweets	4338	2410	2892
Number of words	10892	5769	8859
Average words in each tweet	10.27	7.97	8.20
Avg. characters in each tweet	102.04	49.02	62.50

3.2 Data Preprocessing

Data preparation is crucial in text analytics applications because it filters out irrelevant information from the corpus. In this effort, we execute several preparatory procedures. All uppercase letters are replaced with lowercase ones. Texts containing stop words, numerals, punctuation, and special symbols not needed for sentiment analysis are eliminated. Stop words, such as "a," "an," "the," etc., are crucial in syntax but not meaning. Stemming converts words into lemmas. Stemming converts words into related lemmas.
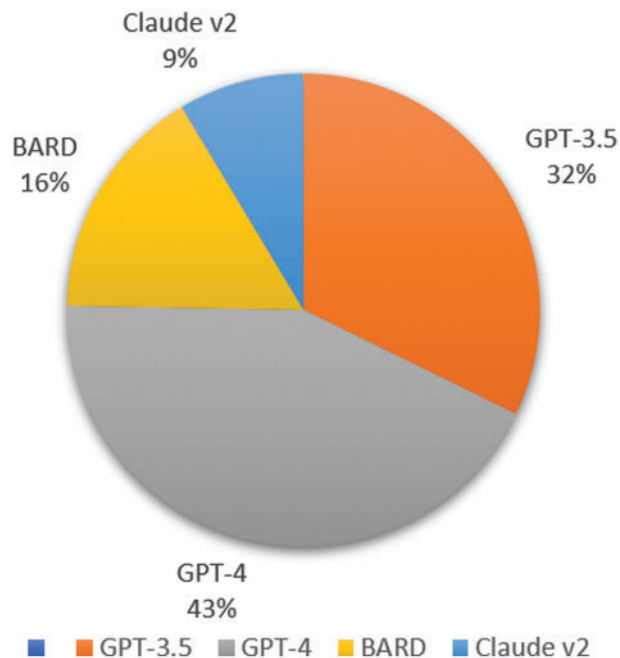
Figure 2: LLM sentiment dataset frequency

3.3 Lexicon-Based Approaches

Lexicon-based or rule-based approaches are a class of methods used in NLP and sentiment analysis to determine the sentiment or polarity of a piece of text based on predefined word lists or lexicons [43]. These methods use sentiment lexicons and dictionaries with words and their related sentiment scores or polarities (positive or negative). We employed the two algorithms, Valence Aware Dictionary and sEntiment Reasoner (VADER), to assess text polarity and conduct a semantic analysis.

Specifically used for sentiment analysis is a rule-based lexicon and analysis method. It is used to extract feelings expressed in social media and performs in this regard. The fundamental basis of VADER sentiment analysis [44,45] is a set of essential elements, including degree modifiers, capitalization, punctuation, conjunctions, and preceding trigrams. VADER categorizes behaviours as positive or negative and produces effective results by developing each word's polarity ratings in the lexicon and standardizing in the range $(-1, 1)$, with "1" being the most positive and "-1" the most negative. The text is negative if the aggregate score is less than -0.05 and positive if it is greater than 0.05. One significant advantage of VADER is that it can generate sentiment polarity straight from the raw tweets without data processing. Additionally, it supports emojis for identifying emotions and is quick enough to be utilized online without degrading speed performance.

3.4 DeBERTa-GRU

The DeBERTa-GRU model, a suggested hybrid of the Gated Recurrent Unit (GRU) [41] and Decoding-enhanced BERT with disentangled attention (DeBERTa) [35], is depicted in Fig. 3.
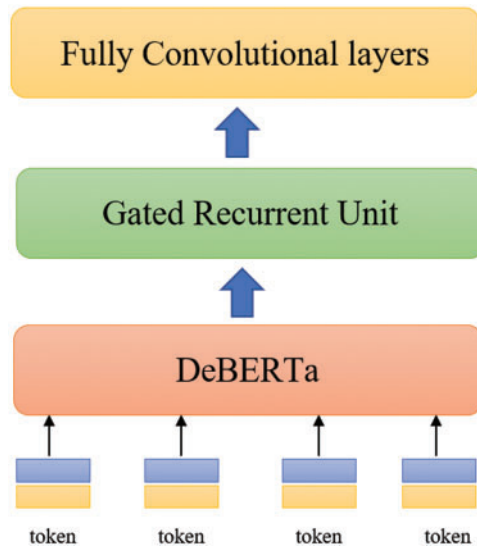
Figure 3: The architecture of the proposed DeBERTa-GRU model

3.4.1 DeBERTa Method

The Bidirectional Encoder Representations from Transformers (BERT) [46] is a language representation model built on top of the Transformer architecture [47], which can be used to enhance the BERT and DeBERTa models. The Transformer architecture's encoder component, which transforms text sequences into rich numerical representations, is the only component used by encoder-only models like BERT [48]. This format makes these models ideal for text classification. BERT's calculated representation should include a token's left and proper contexts. People often call this "bidirectional attention".

A methodically enhanced BERT adaptation is the robustly optimized BERT methodology DeBERTa [49]. To improve BERT's performance, the robustly optimized BERT approach DeBERTa model revised the pre-training technique and trained on larger batches of data.

Two architectural changes are made by decoding improved BERT with disentangled attention DeBERTa [29]. Two vectors are used to encode the meaning and location of each word in a disentangled attention mechanism. H_i denotes the token's contents located at index i. In contrast, P_i represents the relative positioning of the tokens at indexes i and j concerning one another, as mentioned in Eq. (1). The cross-attention score equation is as follows:

$$A_{i,j} = H_i H_j^T + H_i P_{j|i}^T + P_{i|j} H_j^T + P_{i|j} P_{j|i}^T \tag{1}$$

The second change is a better mask decoder that uses absolute values in the decoding layer to predict masked tokens during pre-training. After the transform layers and before the SoftMax layer, DeBERTa combines the final position for masked token prediction. In contrast, the position embedding is a part of the input layer in BERT. Hence, DeBERTa can record the relative position in each transformer layer.

3.4.2 GRU Method

The DeBERTa model uses a GRU [41] to effectively encapsulate the encoding that has long-range connections. The GRU was made to solve the problem of disappearing gradients in RNNs [41]. The GRU's update gate and reset gate are its two primary components. The reset gate regulates how much previous information should be forgotten, whereas the update gate decides which data should be preserved. The input x_t creates outputs for the update gate z_t, reset gate r_t, memory cell c_t, and hidden state h_t at time step t. Calculate these numbers using Eq. (2).

$$z_t = \sigma\ (W_z x_t + U_z h_{t-1} + b_z)$$

$$r_t = \sigma\ (W_r x_t + U_r h_{t-1} + b_r)$$

$$c_t = \tanh\ (W_c x_t + U_c\ (r_t * h_{t-1}) + b_c) \tag{2}$$

$$h_t = z_t * h_{t-1} + (1 - z_t) * c_t.$$

The sigmoid function is denoted by σ and the element-wise multiplication by $*$ in the equations above. The subscripts (W_z, W_r, W_c) and (U_z, U_r, U_c), respectively, designate the update gate, reset gate, and memory cell, and they reflect the weights that must be multiplied with the input and hidden states of h_{t-1}, correspondingly. The biases of the memory cell, the reset gate, and the update gate are written as (bz, br, bc).

The sigmoid function controls the range of the output values of the update and reset gates to be [0, 1]. The memory cell uses the reset gate to decide if the prior information should be kept. Most earlier data is erased when the reset gate's value is near zero.

The current memory cell, affected by the update gate and the previous hidden state, generates the hidden state. When the update gate's value is very close to zero, the contents of the current memory cell are used to replace the secret state. In contrast, most of the information in the preceding cell is discarded. The GRU layer then sends its data to the simplified layer.

3.4.3 Model Training Parameters

The categorical cross-entropy loss function is frequently employed in multiclass classification problems and is used to train the DeBERTa-GRU model. The flat cross-entropy loss quantifies differences between observed and predicted label distributions. Let the accurate label distribution be y_i, and the expected probability distribution for the ith sample be y_i as Eq. (3). Then, we can say the following about the categorical cross-entropy loss function:

$$L = -\frac{1}{N}\sum_{i=1}^{N} y_i \log \hat{y}_i, \tag{3}$$

where N represents the number of dataset samples.

The optimization process aims to update the model parameters θ so that the category cross-entropy loss function is as small as possible.

Combining the strengths of the Nesterov momentum and the Adam optimizers, the DeBERTa-GRU model is optimized with the Adam optimizer. By incorporating gradient information from the following iterations, the Nesterov momentum technique accelerates the convergence rate of

optimization processes. Nadam optimization uses Eq. (4) to modify the model parameters.

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$$
$$n_t = \beta_2 n_{t-1} + (1 - \beta_2) g_t^2$$
$$\hat{m}_t = m_t / (1 - \beta_1^t)$$
$$\hat{n}_t = n_t / (1 - \beta_2^t)$$
$$\theta_{t+1} = \theta_t - \frac{\alpha}{\sqrt{\hat{n}} + e} \hat{m}_{t,}$$

(4)

where g_t is the gradient of the loss function at time step t for model parameters, the first and second-moment represented by m_t and n_t, the exponential decay rates for the first and second-moment denoted by β_1 and β_2, α is the learning rate, the bias-corrected first and second-moment expressed by $\hat{m}_t$ and $\hat{n}_t$, respectively, to avoid division by zero we used ϵ as a small. To arrive at the optimal solution, the Adam optimizer modifies model parameters more rapidly than conventional optimization methods.

4 Result Analysis

In this part, the suggested DeBERTa-GRU model's performance against cutting-edge techniques is compared in a few experimental circumstances. Specifically, it is on the influence of GRU units on model performance while keeping the optimizer fixed at Stochastic Gradient Descent (SGD) and the learning rate at 0.0001.

The maximum number of training epochs is 100. However, an early halting mechanism is used to avoid the overfitting issue. The patience is set to 30 epochs, and the early stopping observation measure is set to validation accuracy. The datasets were split 6:2:2 in all studies for training, testing, and validating. The size of each batch is set to 32. It demonstrates that varying the number of GRU units has a noticeable effect on the model's accuracy, with the highest accuracy of 96.89% achieved when using 256 GRU units, as shown in Table 3.

Table 3: Different learning rates (GRU unit D 256, optimizer D Adam)

Learning rate	Accuracy (%)
0.01	62.83
0.001	85.67
0.0001	96.89
0.00001	89.87

4.1 VADER Analysis

This section discusses the findings of a sentiment analysis conducted on Twitter using VADER's technologies. According to the VADER sentiment analyzer, Fig. 4 displays the sentiment score for each tweet as either positive or negative. After the limitations were applied, categorizing the tweets as positive, negative, and neutral, as shown in Fig. 4, demonstrates how we might classify tweets directly as positive, negative, or neutral using VADER if we selected the proper threshold value.
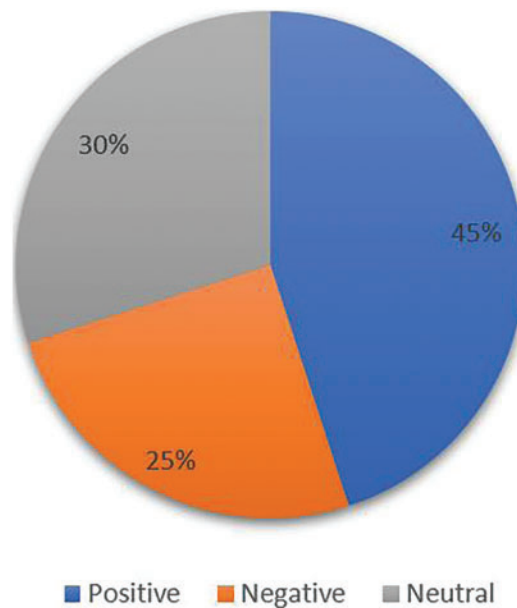
Figure 4: Positive, negative, and neutral tweet classification results from VADER

However, curiously, as seen in Fig. 4, 45% of the tweets conveyed satisfactory thoughts, while just 25% said the opposite, and 30% of the tweets were neutral. Comparing positive and negative tweets, the positive proportion is higher. Because the tiny number of tweets produced imbalanced data and the belief that the threshold value may provide a significant number of neutral viewpoints among all other classes. The results of this research point to the potential application of the VADER Sentiment Analysis to evaluate feelings expressed in tweets and categorize them appropriately, generating accurate results.

4.2 Proposed Models Analysis

This research used four classifiers, Naive Bayes, LSTM, GRU, and DeBERTa-GRU, to assess and contrast their performance in predicting sentiment in Twitter comments related to LLM. A dataset of 9,640 tweets from diverse users was gathered, with each tweet labelled as positive, negative, or neutral using VADER to ensure accuracy and reliability—Table 4 displays status updates within each class, providing a sample of sentiments encountered during the study. The study aimed to reveal the comparative strengths and weaknesses of Naive Bayes, LSTM, GRU, and DeBERTa-GRU classifiers, allowing insights into their efficacy in sentiment analysis for LLM-related content on Twitter.

Table 4: Selected tweets from random users

Tweets	Sentiment analysis
Bard is making false calculations. It can't read the links given, but it is calming to analyze the timings of YouTube videos and provide a time to complete the playlist 😭.	Negative
Google's Artificial Intelligence (AI) chatbot Bard has just debuted in the EU, establishing itself as a direct rival to ChatGPT.	Neutral

(Continued)

Table 4 (continued)

Tweets	Sentiment analysis
I got an invitation from Google to use the new Bard 😑.	Neutral
GPT-4 does terribly when it tries to answer immediately (and then must justify the wrong answer). Better prompting helps a lot; it even gets a correct formula 🩶.	Positive
I believe GPT-4 is a solid enough base to build far more advanced systems than we have to date 🎛️.	
Uploading more significant documents to Claude 2 makes it so valuable for generating summaries 🐿️.	Positive
I'm testing Cohere AI in Code GPT, and it's working great! 🤍	Positive
I tested Bard several months ago, and I think Bard can't answer expert-level questions. Now it's my second time. I must say Bard still can't answer questions like an expert, actually not at an average level 🤍.	Negative

The experiments include Naive Bayes, Gated Recurrent Unit (GRU), Long Short-Term Memory (LSTM), and Decoding-enhanced BERT with disentangled attention-Gated Recurrent Unit (DeBERTa-GRU) as well as other ML and deep learning techniques for sentiment analysis. The experimental findings using the Twitter LLM sentiment dataset are shown in Table 5.

Table 5: The experimental results on the Twitter LLM dataset

Methods	Accuracy	Precision	Recall	F1-score
Naive Bayes	87.93	88	88	88
GRU	90.78	91	91	91
LSTM	91.71	92	92	92
BERT-GRU	90.53	91	91	91
RoBERTa-GRU	93.10	93	93	93
DistilBERT-GRU	92.63	93	93	93
DeBERTa-GRU	96.87	97	97	97

Recall: Classifier-labeled positive samples are the recall. Calculated as:

$$Recall = \frac{TP}{TP + FN} \tag{5}$$

Precision: Is the ratio of correctly categorized positive samples to projected positive samples. It proves positive prediction accuracy, which is calculated as:

$$\Pr ecision = \frac{TP}{TP + FP} \tag{6}$$

F1-score: Recall and precision are weighted in F-score. The F1-score, or F-measure accuracy, is determined as follows:

$$F_1 - score = \frac{P * R}{P + R} * 2 \tag{7}$$

Accuracy: A standard metric is accuracy, which is the percentage of correct predictions. That is:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{8}$$

where TP = True Positive, TN = True Negative, FP = False Positive, FN = False Negative and P = Precision, R = Recall.

4.2.1 Naive Bayes Method

For the Naive Bayes method, our model achieved an accuracy of 87.93%. This metric indicates the overall correctness of our predictions. Moreover, both precision and recall are also at 88%. Precision signifies the ratio of correctly predicted positive observations to the total positive observations, and recall measures the ratio of correctly predicted positive observations to all actual positive observations. These values reflect the Naive Bayes method's balanced performance in identifying positive cases. The F1-score, which combines precision and recall, is also 88%, confirming the method's effectiveness in maintaining a balance between precision and recall.

4.2.2 GRU Method

The GRU method exhibits an accuracy of 90.78%, indicating an improvement over the Naive Bayes approach. The precision, recall, and F1-score are all at 91%, highlighting the consistency of this method in correctly identifying positive cases and achieving a balance between precision and recall.

4.2.3 LSTM Method

The LSTM method further enhances the performance, with an accuracy of 91.71%. It achieves a precision, recall, and F1-score of 92%, indicating even better accuracy and a balanced trade-off between precision and recall compared to the previous methods.

4.2.4 DeBERTa-GRU Method

The DeBERTa-GRU method outperforms all other methods with an impressive accuracy of 96.87%. It also achieves a precision, recall, and F1-score of 97%, indicating superior performance in correctly classifying positive cases and maintaining a high level of precision and recall. The results of the experiments show that the suggested DeBERTa-GRU model is better than the current methods. Our model 'DeBERTa-GRU' model achieved an accuracy of 96.87%, which is significantly higher than the Naive Bayes (87.93%), GRU (90.78%), and LSTM (91.71%) models. This indicates that 'DeBERTa-GRU' consistently makes more correct predictions.
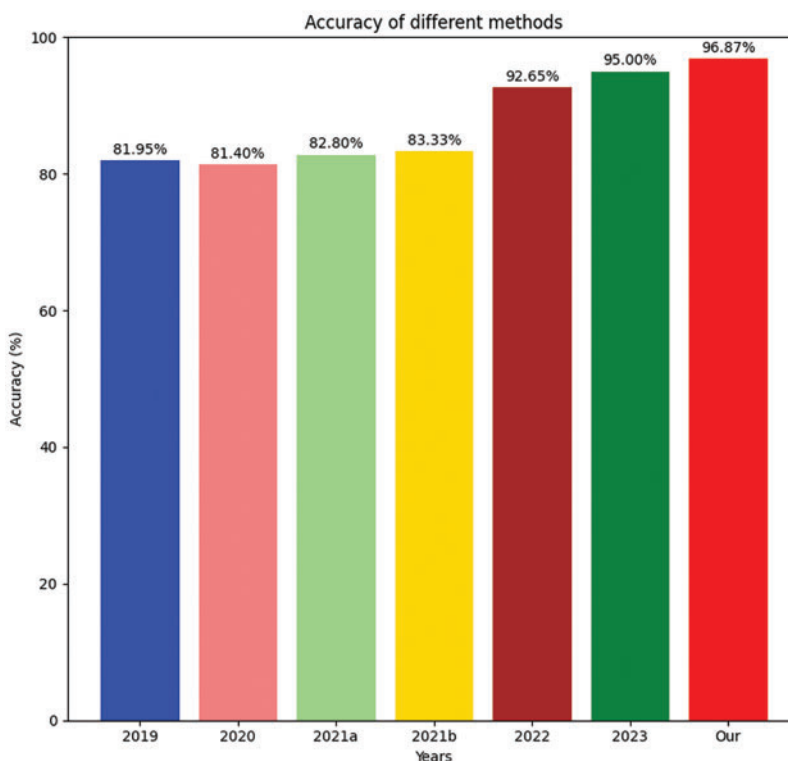
4.3 Comparison with Stat-of-Art Methods

Table 6 presents a comprehensive overview of state-of-the-art methods, showing their corresponding accuracy percentages. The first, "Support Vector Regression", shows 82.95% accuracy, and the last, " Gradient Boosting", has a greater accuracy than other methods at 95.00%. Finally, our proposed "DeBERTa-GRU" method outperformed the previous methods, securing an impressive accuracy

of 96.87%. The table provides a chronological progression of methods, each contributing to the refinement and enhancement of accuracy over the years, as shown in Fig. 5.

Table 6: Result comparison with stat-of-art methods

Methods	Years	Acc. (%)
Support vector regression [50]	2019	81.95
TF-IDF + logistic regression [51]	2020	81.40
Word2Vec + BFTAN [52]	2021	82.80
RoBERTa [53]	2021	83.33
LSTM + FastText, BERT [54]	2022	92.65
Gradient boosting [55]	2023	95.00
DeBERTa-GRU (Our)	–	**96.87**

Figure 5: Accuracy comparison with stat-of-art

5 Conclusion

A powerful sentiment analysis tool is crucial in significant data growth in various fields, particularly social and economics. The sentiment analysis's input drives the decision-making of the interested parties. Most currently available sentiment analysis research uses ML and deep learning techniques,

i.e., SVM RNN. In previous studies, researchers used Transformers to do sentiment analysis. Therefore, the DeBERTa-GRU model, which is a combination of the Transformer and RNN models, is presented in this study. The first step is data preparation, which involves standardizing the text and eliminating unnecessary words. The resulting clean corpus is then used for training and sentiment analysis using the suggested DeBERTa-GRU model. The proposed DeBERTa-GRU model uses the best parts of both DeBERTa and GRU. For example, DeBERTa fast word embedding encoding works well with GRU's understanding of distance relationships. The results of the experiments show that the suggested DeBERTa-GRU model is better than the current best methods for sentiment analysis on the Twitter LLM dataset. Based on what was found, most Twitter users like using Chatbot.

Author Contributions: Methodology and formal analysis: A. Assiri, A. Gumaei; project administration and data curation: F. Mehmood; the final revision and English polishing: T. Abbas; model analysis and result correction: Sami Ullah. All authors have read and agreed to the published version of the manuscript.

Availability of Data and Materials: The codes and data are available under request from the authors.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1]   S. Tam, R. B. Said, and Ö. Ö. Tanriöver, "A ConvBi-LSTM deep learning model-based approach for Twitter sentiment classification," *IEEE Access*, vol. 9, pp. 41283–41293, Mar. 2021. doi: 10.1109/ACCESS.2021.3064830.

[2]   K. Ullah, I. Mumtaz, M. A. Zia, and A. Razzaq, "Text based emotion detection by using classification and regression model," presented at the 2022 Int. Conf. Manag. Sci. Engi. Manag., New York, NY, USA, Dec. 15–20, 2022, pp. 414–419.

[3]   T. K. Long, C. P. Lee, K. S. M. Anbananthen, and K. M. Lim, "RoBERTa-LSTM: A hybrid model for sentiment analysis with transformer and recurrent neural network," *IEEE Access*, vol. 10, pp. 21517–21525, Jan. 2022. doi: 10.1109/ACCESS.2022.3152828.

[4]   C. Ying, Z. Sun, L. Li, and W. Mo, "A study of sentiment analysis algorithms for agricultural product reviews based on improved BERT model," *Symmetry*, vol. 14, no. 8, pp. 1604, Aug. 2022. doi: 10.3390/sym14081604.

[5]   J. J. Hassannataj *et al.*, "BERT-deep CNN: State of the art for sentiment analysis of COVID-19 tweets," *Soc. Net. Ana. Mini.*, vol. 13, no. 1, pp. 99, Nov. 2023. doi: 10.1007/s13278-023-01102-y.

[6]   M. M. Erfan, M. S. Abadeh, and H. Keshavarz, "Aspect-based sentiment analysis using adaptive aspect-based lexicons," *Expert Syst. Appl.*, vol. 148, pp. 113234, Jun. 2020. doi: 10.1016/j.eswa.2020.113234.

[7]   N. K. Ke and V. Uma, "Intelligent sentinet-based lexicon for context-aware sentiment analysis: Optimized neural network for sentiment classification on social media," *J. Sup. Comp.*, vol. 5, pp. 1–25, Apr. 2021. doi: 10.1007/s11227-021-03709-4.

[8]   K. Ramandeep and S. Kautish, "Multimodal sentiment analysis: A survey and comparison," *Res. Anth. Imple. Senti. Anal. Acro. Multi. Discip.*, vol. 4, pp. 1846–1870, Jun. 2022. doi: 10.4018/978-1-6684-6303-1.ch098.

[9] V. Sanjeev, "Sentiment analysis of public services for smart society: Literature review and future research directions," *Govt Inform. Quart.*, vol. 39, no. 3, pp. 101708, Jul. 2022. doi: 10.1016/j.giq.2022.101708.

[10] J. E. Vahlne and J. Johanson, "The Uppsala model: Networks and micro-foundations," *J. Int. Bus.*, vol. 51, pp. 4–10, Nov. 2020. doi: 10.1057/s41267-019-00277-x.

[11] B. Xu, C. Gil-Jardiné, F. Thiessard, E. Tellier, M. Avalos and E. Lagarde, "Pre-training a neural language model improves the sample efficiency of an emergency room classification model," presented at the 2020 Int. Fla. Conf., North Miami Beach, Florida, USA, May 17–20, 2020, pp. 538–543.

[12] Z. Ali, A. Razzaq, S. Ali, S. Qadri, and A. Zia, "Improving sentiment analysis efficacy through feature synchronization," *Multimed. Tools Appl.*, vol. 80, pp. 13325–13338, Apr. 2021. doi: 10.1007/s11042-020-10383-w.

[13] P. Verga, H. Sun, L. B. Soares, and W. Cohen, "Adaptable and interpretable neural memoryover symbolic knowledge," presented at the 2021 Conf. N.A. Chapt. Associ. Comput. Ling. Hum. Lang. Tech., Mexico, USA, Jun. 1–5, 2021, pp. 3678–3691.

[14] L. Congmei *et al.*, "Tuning the mechanical performance efficiently of various LLM-105 based PBXs via bioinspired interfacial reinforcement of polydopamine modification," *Compos B. Eng.*, vol. 186, pp. 107824, Apr. 2020. doi: 10.1016/j.compositesb.2020.107824.

[15] S. A. Alanazi *et al.*, "Public's mental health monitoring via sentimental analysis of financial text using machine learning techniques," *Int. J. Environ. Res. Public Health*, vol. 19, no. 15, pp. 9695, Jul. 2022. doi: 10.3390/ijerph19159695.

[16] M. Rehman *et al.*, "Semantics analysis of agricultural experts' opinions for crop productivity through machine learning," *Appl. Artif. Intell.*, vol. 36, no. 1, pp. 2012055, Dec. 2022. doi: 10.1080/08839514.2021.2012055.

[17] M. Aljanabi, M. Ghazi, A. H. Ali, and S. A. Abed, "ChatGPT: Open possibilities," *Iraqi J. Comput. Sci. Math.*, vol. 4, no. 1, pp. 62–64, Jan. 2023. doi: 10.52866/ijcsm.2023.01.01.0018.

[18] H. Hassani and E. S. Silva, "The role of ChatGPT in data science: How ai-assisted conversational interfaces are revolutionizing the field," *Big Data Cogn. Comput.*, vol. 7, no. 2, pp. 1–16, Mar. 2023. doi: 10.3390/bdcc7020062.

[19] G. Xiao, J. Lin, M. Seznec, H. Wu, J. Demouth and S. Han, "Smoothquant: Accurate and efficient post-training quantization for large language models," presented at the 2023 Int. Conf. Mac. Lear., Honolulu, Hawaii, USA, Jul. 23–29, 2023, pp. 38087–38099.

[20] M. Wongkar and A. Angdresey, "Sentiment analysis using Naive Bayes algorithm of the data crawler: Twitter," presented at the 2019 Int. Conf. Inform. Compt., Semarang, Indonesia, Oct. 16–17, 2019, pp. 1–5.

[21] F. Mehmood, E. Chen, M. A. Akbar, and A. A. Alsanad, "Human action recognition of spatiotemporal parameters for skeleton sequences using MTLN feature learning framework," *Electronics*, vol. 10, no. 21, pp. 2708, Oct. 2021. doi: 10.3390/electronics10212708.

[22] A. M. Rahat, A. Kahir, and A. K. M. Masum, "Comparison of Naive Bayes and SVM algorithm based on sentiment analysis using review dataset," presented at the 2019 Int. Conf. Syst. Mode. Advanc. Res. Tre., Moradabad, India, Nov. 22–23, 2019, pp. 266–270.

[23] D. K. Madhuri, "A machine learning based framework for sentiment classification: Indian railways case study," *Int. J. Innov. Technol. Expl. Eng.*, vol. 8, no. 4, pp. 441–445, Feb. 2019. doi: 10.1007/978-981-15-7062-9_75.

[24] E. Prabhakar, M. Santhosh, A. H. Krishnan, T. Kumar, and R. Sudhakar, "Sentiment analysis of US airline twitter data using new adaboost approach," *Int. J. Eng. Res. Technol.*, vol. 7, no. 1, pp. 1–6, Dec. 2019. doi: 10.17577/IJERTCONV7IS01003.

[25] A. I. Saad, "Opinion mining on US Airline Twitter data using machine learning techniques," presented at the 2020 Int. Conf. Comp. Eng., Cairo, Egypt, Dec. 29–30, 2020, pp. 59–63.

[26] A. Younas, R. Nasim, S. Ali, G. Wang, and F. Qi, "Sentiment analysis of code-mixed Roman Urdu-English social media text using deep learning approaches," presented at the 2020–2021 Int. Conf. Comput. Sci. Eng., Guangzhou, China, Dec. 29–Jan. 1, 2020–2021, pp. 66–71.

[27] S. Anbukkarasi and S. Varadhaganapathy, "Analyzing sentiment in Tamil tweets using deep neural network," presented at the 2020 Int. Conf. Comp. Method. Commu., Erode, India, Mar. 11–13, 2020, pp. 449–453.

[28] N. K. Thinh, C. H. Nga, Y. S. Lee, M. L. Wu, P. C. Chang and J. C. Wang, "Sentiment analysis using residual learning with simplified CNN extractor," presented at the 2020 Int. Conf. Sympo. Multi., San Diego, CA, USA, Dec. 9–11, 2020, pp. 335–3353.

[29] D. A. Kumar and A. Chinnalagu, "Sentiment and emotion in social media COVID-19 conversations: SAB-LSTM approach," presented at the 2020 Int. Conf. Syst. Mode. Advanc. Res. Tre., Moradabad, India, Dec. 4–5, 2020, pp. 463–467.

[30] K. Dhola and M. Saradva, "A comparative evaluation of traditional machine learning and deep learning classification techniques for sentiment analysis," presented at the 2020 Int. Conf. Clo. Compt., Dat. Sci. Eng., Noida, India, Jan. 28–29, 2020, pp. 932–936.

[31] A. H. Uddin, D. Bapery, and A. S. M. Arif, "Depression analysis from social media data in Bangla language using long short-term memory (LSTM) recurrent neural network technique," presented at the 2019 Int. Conf. Compt. Commu. Chem. Mate. Elect. Eng., Rajshahi, Bangladesh, Jul. 11–12, 2019, pp. 1–4.

[32] F. Mehmood, E. Chen, T. Abbas, M. A. Akbar, and A. A. Khan, "Automatically human action recognition (HAR) with view variation from skeleton means of adaptive transformer network," *Soft Comput.*, vol. 7, pp. 1–20, Apr. 2023. doi: 10.1007/s00500-023-08008-z.

[33] R. Alkanhel, A. Ali, F. Jamil, M. Nawaz, F. Mehmood and A. Muthanna, "Intelligent transmission control for efficient operations in SDN," *Comput. Mater. Contin.*, vol. 71, no. 2, pp. 2807–2825, Dec. 2022. doi: 10.32604/cmc.2022.019766.

[34] R. M. Alahmary, H. Z. Al-Dossari, and A. Z. Emam, "Sentiment analysis of Saudi dialect using deep learning techniques," presented at the 2019 Int. Conf. Elect. Infor. Commu., Auckland, New Zealand, Jan. 22–25, 2019, pp. 1–6.

[35] Y. G. Jung, K. T. Kim, B. Lee, and H. Y. Youn, "Enhanced Naive Bayes classifier for real-time sentiment analysis with SparkR," presented at the 2016 Int. Conf. Infor. Commun. Tech. Conv., Jeju Island, South Korea, Oct. 19–21, 2016, pp. 141–146.

[36] B. Vadim, T. Leemann, K. Seßler, J. Haug, M. Pawelczyk and G. Kasneci, "Deep neural networks and tabular data: A survey," *IEEE Trans. Nurl. Net. Lear. Syst.*, pp. 1–21, Nov. 2022. doi: 10.1109/TNNLS.2022.3229161.

[37] B. Yoshua, R. Ducharme, and P. Vincent, "A neural probabilistic language model," *Advanc. Nurl Inform. Proce. Syst.*, vol. 13, pp. 400–406, Aug. 2000.

[38] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng and C. Potts, "Learning word vectors for sentiment analysis," presented at the 2011 Int. Conf. Annl. Meet. Assoc. Comput. Ling., Portland, Oregon, USA, Jun. 19–24, 2011, pp. 142–150.

[39] L. Igor and H. Lipson, "Re-embedding words," presented at the 2013 Int. Conf. Annl. Meet. Associ. Compt. Ling., Sofia, Bulgaria, Aug. 4–9, 2013, pp. 489–493.

[40] T. Duyu, F. Wei, B. Qin, N. Yang, T. Liu and M. Zhou, "Sentiment embeddings with applications to sentiment analysis," *IEEE Trans. Knowlg. Data Engin.*, vol. 28, no. 2, pp. 96–509, Oct. 2015. doi: 10.1109/TKDE.2015.2489653.

[41] Q. Li, "Predicting effective arguments based on a hybrid language model," presented at the 2023 Int. Conf. Compt. Art. Intel. Cont. Eng., Hangzhou, China, May 23, 2023, pp. 949–953.

[42] K. L. Tan, C. P. Lee, and K. M. Lim, "RoBERTa-GRU: A hybrid deep learning model for enhanced sentiment analysis," *Appl. Sci.*, vol. 13, no. 6, pp. 3915, 2023. doi: 10.3390/app13063915.

[43] M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede, "Lexicon-based methods for sentiment analysis," *Comput. Linguist.*, vol. 37, no. 2, pp. 267–307, Jun. 2011. doi: 10.1162/COLI_a_00049.

[44] R. D. Endsuy, "Sentiment analysis between VADER and EDA for the US presidential election 2020 on twitter datasets," *J. Appl. Data Sci.*, vol. 2, no. 1, pp. 8–18, Jan. 2021. doi: 10.47738/jads.v2i1.17.

[45] A. Razzaq *et al.*, "Extraction of psychological effects of COVID-19 pandemic through topic-level sentiment dynamics," *Complexity*, vol. 2022, pp. 1–10, Mar. 2022. doi: 10.1155/2022/9914224.

[46] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," presented at the 2019 Conf. North Ameri. Chapt. of the Asso. for Comp. Ling.: Hum. Lang. Tech., 2019, vol. 1, pp. 4171–4186.

[47] B. Jin, Y. Zhang, Q. Zhu, and J. Han, "Heterformer: Transformer-based deep node representation learning on heterogeneous text-rich networks," presented at the 2023 Int. Conf. Knowl. Disco. Dat. Mini., Long Beach, California, USA, Aug. 06–10, 2023, pp. 1020–1031.

[48] K. Wang, S. C. Han, and J. Poon, "InducT-GCN: Inductive graph convolutional networks for text classification," presented at the 2022 Int. Conf. Patt. Reco., Montreal, QC, Canada, Aug. 21–25, 2022, pp. 1243–1249.

[49] F. Mai, L. Galke, and A. Scherp, "Using deep learning for title-based semantic subject indexing to reach competitive performance to full-text," presented at the 2018 Int. Conf. Digt. Libra., Fort Worth, TX, USA, Jun. 3–7, 2018, pp. 169–178.

[50] S. Elbagir and J. Yang, "Twitter sentiment analysis based on ordinal regression," *IEEE Access*, vol. 7, pp. 163677–163685, May 2019. doi: 10.1109/ACCESS.2019.2952127.

[51] K. Chakraborty, S. Bhatia, S. Bhattacharyya, J. Platos, R. Bag and A. E. Hassanien, "Sentiment analysis of COVID-19 tweets by deep learning classifiers—A study to show how popularity is affecting accuracy in social media," *Appl. Soft Comput.*, vol. 97, pp. 106754, Dec. 2020. doi: 10.1016/j.asoc.2020.106754.

[52] S. M. Alhashmi, A. M. Khedr, I. Arif, and M. El Bannany, "Using a hybrid-classification method to analyze Twitter data during critical events," *IEEE Access*, vol. 9, pp. 141023–141035, 2021. doi: 10.1109/ACCESS.2021.3119063.

[53] J. Choudrie, S. Patil, K. Kotecha, N. Matta, and I. Pappas, "Applying and understanding an advanced, novel deep learning approach: A COVID-19, text-based, emotions analysis study," *Inf. Syst. Front.*, vol. 23, no. 6, pp. 1431–1465, Dec. 2021. doi: 10.1007/s10796-021-10152-6.

[54] A. Alsayat, "'Improving sentiment analysis for social media applications using an ensemble deep learning language model," *Arabian J. Sci. Eng.*, vol. 47, no. 2, pp. 2499–2511, Oct. 2022. doi: 10.1007/s13369-021-06227-w.

[55] A. M. Shahedul *et al.*, "Harmonizing macro-financial factors and Twitter sentiment analysis in forecasting stock market trends," *J. Comp. Sci. Tech. Stud.*, vol. 6, no. 1, pp. 58–67, Jan. 2024. doi: 10.32996/jc-sts.2024.6.1.7.