Proceedings of the Fourth International Conference on Electronics and Sustainable Communication Systems (ICESC-2023)
IEEE Xplore Part Number: CFP23V66-ART; ISBN: 979-8-3503-0009-3

2023 4th International Conference on Electronics and Sustainable Communication Systems (ICESC) | 979-8-3503-0009-3/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICESC57686.2023.10193121

Text Sentiment Analysis of Film Reviews Using Bi-LSTM and GRU

K. Mouthami
Assistant Professor
Dept. of Computer Science Engineering
KPR Institute of Engineering and Technology
Coimbatore, India
mouthami.k@kpriet.ac.in

N. Yuvaraj
Professor
Dept. of Computer Science Engineering
KPR Institute of Engineering and Technology
Coimbatore, India
yuvaraj.n@kpriet.ac.in

K.K. Thilaheswaran
Student, Dept. of Artificial Intelligence and Data Science
KPR Institute of Engineering and Technology
Coimbatore, India
20ad058@kpriet.ac.in

K.J. Lokeshvar
Student, Dept. of Artificial Intelligence and Data Science
KPR Institute of Engineering and Technology
Coimbatore, India
21adl03@kpriet.ac.in

***Abstract*— Sentiment analysis is a vast subject to explore in natural language processing (NLP) techniques. The film reviews were analyzed and segregated into positive, neutral, and negative reviews. The proposed model examines two distinct datasets, one with multi-class labels and the other with binary-type class labels. The preprocessing is done by using the bag of words and skip-gram word2vec, variety of classifiers, including Two state Gated recurrent units (TS-GRU), was used for binary classification, and the Bidirectional Long Short-Term Memory (Bi-LSTM) for the multi-class situation was implemented. Bi-LSTM was utilized in the model, in which symmetric Bi-LSTM replaces the LSTM approaches to get around the high computational cost of training the normal LSTM. The Bi-LSTM has equivalent accuracy to the LSTM while significantly reducing computational costs. Sentiment analysis offers a glimpse of how consumers feel about certain movies. This endeavor is undertaken via the study of natural language processing, in which sentiment analysis is a branch which examines how words are arranged and utilized to extract meaning from user writing. This helps to investigate further the methods for deciphering such data thanks to significant advancements in machine learning techniques; with Keras API and the Internet Movie Database (IMDb) dataset, this study compares single and multi-branch CNN with Bidirectional LSTMs with different kernel sizes. GRU achieved 98.24% of accuracy, and bi-LSTM achieved 98.65% of accuracy.**

Keywords*— *Natural language processing, Machine learning, Bidirectional long short-term memory, Gated recurrent units, Sentiment analysis, movie review.

I. INTRODUCTION

Sentiment analysis is the standard process that takes place in natural language processing. The task is to identify the polarity of the text in the given group of texts. It thoroughly examines the many techniques, standards, and resources used in sentiment analysis and opinion mining [1]. Several types of emotions can exist. Two instances in this study are as follows: A movie review might be positive (+) or negative (-) [2]. This is comparable to the model using an advanced similarity metric by analyzing the text's emotion after being summarized. A movie review may be neutral, somewhat positive, highly positive, somewhat negative, neutral, or very positive [3].

There are two critical components to the task. The model tests several fundamental sentiment analysis methods in the first section. This is a fair starting point for evaluating more sophisticated techniques. The model tests many variations of the fundamental models in the second section. This segment's goal is to train a binary classifier that will be used to categorize movie reviews into positive and negative categories. The initial step in this problem, like in many other natural language challenges, is to preprocess and transform the words (movie reviews) converted into a numerical form of features. Several techniques can be used to accomplish this, including the bag of words and word-to-vector. After the process of preprocessing and feature extraction, the classifier utilizes the dataset used for the second instance [4]. Each sample in this dataset is broken down using a recursive tree (training and testing examples included) for the classification, and several recursive neural networks (RNN) variants are implemented [5,6]. Here the mentioned problem has multiple classes different from the first class.

Sentiment analysis is one of the most challenging tasks in natural language processing. Sentiment review categorization is most frequently utilized in sentiment analysis [7]. The sentiment classification aims to categorize user opinions, attitudes, and emotions conveyed in text into positive, negative, and neutral polarities based on sentiment. To tackle the sentiment analysis challenge, several cutting-edge deep learning methods have been presented [8]. One of the most extensively used deep learning architectures for sentiment analysis is the recurrent neural network (RNN). The Two State GRU (TS-GRU) is based on the feature attention mechanism suggested in the proposed research. It focuses on word-feature seizing and sequential modelling for identifying and categorizing sentiment polarity. Word2vec's CBOW architecture is the foundation for the Bi-LSTM neural network design [9]. The main goal of the model is to effectively learn a function for embedding generic context that encompasses variable-length sentences around the target word. This simplifies the process of expressing context in a more natural way. Sentence completion, vocabulary replacement, and word sense clarification improved, consequently exceeding prior approaches to the broad contextual representation of typical word embedding. For medical texts, yet another fundamental model was put out separately. The reason is that medical language has most of the technical information that only the subject-matter specialists could grasp, so the model is devised for comprehending complicated, particular medical structures [10]. A specialized generic corpus is used to produce the suggested embedding model, which may or may not include characteristics created by medical professionals. But movie reviews only have layman's terms, so it is easy to work with movie reviews.

II. RELATED WORKS:

An effective text categorization model based on Word2vec and LSTM is suggested for the security industry. To get around the high dimensionality that was a problem with conventional approaches, A pre-trained Word2vec model was employed. Finally, the study accomplished patent text classification in the security sector by extracting text functionalities and training the LSTM classification model [8]. The findings demonstrate that the model can accurately classify these patent papers. This created a strong platform for future study and efficient patent usage. CNN and LSTM have been coupled in several experiments to improve classification accuracy [11]. The LSTM model and CNN were used with unexpectedly positive outcomes for a range of natural language processing applications. A suggested hybrid model employs an extremely deep CNN and LSTM to address sentiment analysis's prior problems to improve prediction accuracy. The hybrid CNN-LSTM model proposed in this study uses dropout, normalization, and rectified linear units. The results indicate that this model outperforms traditional deep learning and machine learning techniques regarding precision, recall, F1-score, and accuracy, as reported in reference [12]. The trials conducted using eight variations of the CNN-LSTM text classification model demonstrate that the accuracy of the combination of CNN and LSTM without an activation function or its derivative was the highest.

A neighborhood CNN-LSTM model with a tree structure to track emotional analysis phases. The proposed regional CNN model differs from conventional CNNs in using a specific section of the text as a region rather than taking the entire text as input. In contrast, conventional CNNs divide the text into multiple regions and extract useful emotional information from each region, weighting them accordingly [13]. The classification accuracy is further improved by combining CNN and LSTM since this combination considers both long-distance relationships between sentences and local (regional) information in sentences. New hybrid CNN-LSTM models have been put out and outperformed earlier ones in terms of performance. A different hybrid strategy that makes use of CNN's ability to extract local characteristics while addressing its intrinsic inability to convey long-term contextual data [14]. The model also aims to address LSTM's inherent flaws, which include its sequential information processing, which makes it the weakest feature extractor. While the hybrid model outperformed its competitors in terms of outcomes, it fell short compared to models that employ an attention mechanism in terms of compelling findings [15]. A brand-new hybrid model that integrates multiple learning methodologies (LSTM, Gated recurrent unit (GRU), Bi-LSTM, CNN) with various word embeddings (Word2Vec, FastText, and character-level embedding). The model extracted features using CNN and LSTM.

The findings have not improved with this hybrid technique since no attention mechanism exists [16]. Modern findings were attained with the introduction of the attention models more recently. A technique that combines label embedding with a self-interaction attention strategy. The model uses the unique BERT (bidirectional encoder representation from transformers) technology for text extraction. The method uses the self-interaction attention mechanism to increase classification accuracy by jointly embedding words and labels. The BERT model is challenging to train, particularly for large text [17]. Traditional machine learning techniques such as decision trees and K Nearest Neighbor (KNN) have demonstrated promising outcomes for specific categorization scenarios. More than a billion words were used to train a Fast Text model, which was then used to classify 50,000 phrases into different categories [18]. This method, which appears conventional, fared better than specific deep learning methods. Even yet, it was still being determined if sentiment categorization could be used as an appropriate comparison tool between shallow models and deep learning techniques, which have the far greater representational ability [19]. The models like Support Vector Machines (SVM), Random Forest, Multinomial Naive Bayes, Long Short-Term Memory LSTM, and CNN are used[20]. The accuracy while developing the model using SVM is 79%[21]. The accuracy of the deep learning model would be 89% for the testing data[22].

TABLE I. LITERATURE SURVEY

Authors	Technique	Performance	Advantage	Disadvantage
Pouransari, Hadi, and Saman Ghili.	The low ranked-Recursive neural tensor network	Usage of multiple classes to decrease	Decreases computational cost	Low accuracy compared to standard RNTN
Pang, Bo, and Lillian Lee.	Sentiment Analysis	Opinion oriented information seeking	Categorization, extraction, and summarization of text data	The traditional fact-based analysis is not much effective
Pang, Bo, and Lillian Lee.	standard multi-class text categorization	Implement of meta-algorithm based on a metric labelling formulation	The performance gains achieved by SVMs with respect to multi-class and regression versions are substantial.	The complexity of the problem is high
Pang, Bo, and Lillian Lee.	Naive Bayes polarity classifier and SVM	Removal of objective sentences	polarity-classification accuracy is high	Parameter selection techniques are not involved
Richard Socher, Alex Perelygin, Jean Y. Wu	Recursive Neural Tensor Network and Sentiment Treebank	each sentence is classified into positive or negative	Helps to increase the accuracy level	Data from social media is not suitable for analyzing
Socher, Richard, Cliff C. Lin, Chris Manning	Recursive neural networks	recovering recursive structure both in complex scene images	Outperform alternative approaches for semantic scene segmentation	Requirement of high computational power
Socher, Richard, Jeffrey Pennington, Eric H. Huang	Sentiment Analysis and Recursive Autoencoder	Predicting sentiment label distributed elements	Accurately predict sentence-level sentiment distributions.	Specific characters cannot be identified to classify
Melamud, Oren, Jacob Goldberger, and Ido Dagan	Bi-directional LSTM	Sentence completion, lexical substitution tasks are the results	The model could be useful in a wide variety of NLP tasks.	Only by using hyperparameters accuracy is increased

Ceraj, Tin, Ivan Kliman, and Mateo Kutnjak	Comparison of two different Word2vec embedding models	Embedding quality comparison is done with a multilayer bi-LSTM model.	Deeper models with conv layers had improved results	Various pre-processing techniques need to be done.
Xiao, Lizhong, Guangzhong Wang, and Yang Zuo.	Classification methods like KNN and LSTM	Training the word vector based on word2vec by the LSTM classification algorithm	Naturally handles multi-class cases.	Easy to overfit while complying
Rehman, Anwar Ur, Ahmad Kamran Malik, Basit Raza, and Waqar Ali	Hybrid CNN-LSTM approach	LSTM model to detect deeper semantics of words and deep CNN model on the supervised dataset	The proposed model also uses dropout technology, normalization	Spatial correlation surpasses the model
Wang, Jin, Liang-Chih Yu, K. Robert Lai, and Xuejie Zhang.	CNN-LSTM Model for Dimensional Sentiment Analysis	Strategy for regional division involves identifying regions at various levels within predetermined parser tree.	Phrases and clauses are identified easily	Discovering linguistic regions are not performed
Xiangyang She, Di Zhang	A hybrid CNN-LSTM model is proposed	CNN is utilized to extract local text features, while LSTM captures historical information.	Reduces dimensionality, enhances learning ability	High complexity by using hybrid models
Küpper, Axel, Younghee Park, Peter Ruppel, Stefan Schulte, and Jie Xu	Deep learning methods (Bi-LSTM, CNN) are used	An ensemble learning model based on soft voting was employed to classify the attributes.	Effective performance of different classifiers on different modalities	Computationally expensive in usage
Salur, Mehmet Umut, and Ilhan Aydin.	Deep learning techniques like Bidirectional LSTM and GRU are used	Higher classification performance implemented	The hybrid model can be improved and enriched by attention mechanisms	Difficult to analyze morphologically
Zhang, Jiarui, Yingxiang Li, Juan Tian, and Tongyan Li..	Combines the advantage of two traditional neural network models (LSTM, CNN)	Performance of the hybrid model is evaluated in comparison to other models.	Improves the accuracy of text classification	The complexity of the model is directly proportional to the amount of text length
Beylkin, Gregory, Jochen Garcke, and Martin J. Mohlenkamp	Multivariate regression model	The process of learning a function of multiple variables from sparse data is tackled.	Avoidance of overfitting	Computational cost to set up these problems
Maas, Andrew, Raymond E. Daly, Peter T. Pham,	Natural language processing techniques	Semantic similarities are identified and polarity classification	The model is highly flexible and performed better than LDA	Suitable only to the unlabelled data

III. Dataset

The labelled dataset comprises the 50,000 IMDB movie reviews specifically chosen for sentiment analysis. Reviews' sentiment is binary, hence if an IMDB rating is below 5, it receives a sentiment value of 0, and if it is more than or equal to 6, The sentiment score for this movie is 1, and no more than 30 individuals have reviewed it. No films in the 25,000-review contained test set are also in the 25,000-review labelled training set, as shown in Table 2. In addition, 50,000 more IMDB reviews are offered without star ratings [20,21]. There are four file descriptions for this particular dataset which are,

- Labelled Train Data
- Test Data
- Un-labelled Train Data
- Sample Submissions

A. *Labelled Train Data*

The tab-delimited file includes a header row of attributes named id, sentiment, and review. Each attribute contains 25,000 values related to their data fields.

B. *Test Data*

There is a header row in the tab-delimited file, then 25,000 rows of text and an id for each review. The test data helps to predict how each one reviews sentiments.

C. *Un-Labelled Train Data*

The tab-delimited file consists of a header row followed by 50,000 rows, each containing an id and text for an individual review, without any accompanying labels.

D. *Sample Submissions*

A noise-free file format of a comma-delimited sample submission is listed in this file description.

Three data fields are mentioned in the dataset: id, sentiment, and review.

1) *ID*
Each review has its unique id.
2) *Sentiment*
The review data is annotated with 1 to indicate a positive review and 0 to indicate a negative review.
3) *Review*
A basic review of the movies is explained in a few words.

TABLE II. DATA PROCESSING RATIO

Training Data	50%
Testing Data	50%

IV. Methodology

As shown below, implemented model tests many fundamental sentiment analysis techniques. The next step is to clean up the data using a pretreatment step for each approach. The Python programme Beautiful Soup can be used to remove HTML elements, extra punctuation can be removed using the Python module Regular Expression, the text may be converted to lowercase, and stop words can be removed if necessary, using the Python Natural Language Toolkit. The model tests the following techniques to turn a cleaned sequence of words into numerical feature vectors:

- Bag of Words (BOW)

The most straightforward approach to mathematically represent texts is using a bag of words. Using a given vector for converting to a text represents the number of times the text has used the word from the vocabulary except for very uncommon terms (using the 5000 most frequent words). The vocabulary size, encompassing all words in the review collection, determines the processing time. Following the generation of BOW vectors for each review in the labelled training set, a classifier is fit to the data.

- Word2Vec

The conversion of each word in the text into a vector is another method for mathematically representing texts. This transformation should preserve the semantics of words. For example, if two words have similar meanings, their vectors should also be similar (in an L2-distance sense). The word2vec task's independence from the main goal (here, sentiment analysis) and lack of dependence on a labelled dataset are significant features [22]. To train word vectors by employing all 75,000 reviews as the corpus. In addition to the standard preprocessing procedures on the raw reviews, it must divide paragraphs into sentences to train word vectors because the word2vec method only takes sentences as input.

- Words to reviews - Clustering

By converting words to vectors, similarities and distances between words can be determined, allowing related terms to be grouped. Using the K-means technique, word clusters are formed with an average size of 5 words per cluster, and a centroid represents each cluster. The review can now be viewed as a collection of centroids, akin to the bag of words. Following the generation of a bag of centroids vectors for each review in the labelled training set, a classifier is trained on the data.

A. *Sentiment Analysis*

Sentiment analysis, also called opinion mining, is a natural language processing (NLP) approach used to determine data's positivity, negativity, or neutrality. Businesses often use sentiment analysis to monitor customer reviews and gauge the perception of their brands and products in their target market. Sentiment analysis is concerned with determining the polarity of text (positive, negative, or neutral), as well as identifying various emotions (such as anger, joy, and sadness), urgency (urgent or not urgent), and even intents (interested or not interested). This method is widely used to analyze social media data, evaluate brand reputation, and gain a deeper understanding of customers. The sentiment analysis may help to construct and customize the categories to match the sentiment analysis needs to be based on the wish of the customization to interpret consumer comments and inquiries.

B. *Recurrent Neural Network*

A recurrent neural network (RNN) is an artificial neural network that specializes in sequential or time series data. RNNs are utilized in popular applications such as Siri, voice search, and Google Translate. Some are commonly used for temporal or sequential problems, such as language translation, natural language processing (NLP), speech recognition, and image captioning. Like feedforward and convolutional neural networks (CNNs), RNNs learn from training data. However, RNNs have a unique "memory" that enables them to use data from previous inputs to affect the current input and output. Unlike traditional deep neural networks, where inputs and outputs are assumed to be independent, the output of RNNs is influenced by the previous elements in the sequence. Unidirectional recurrent neural networks cannot consider future events in their forecasts, although it would be helpful to in deciding the output of a particular sequence.

Recurrent networks employ this information to anticipate the following word in the sequence by considering the order in which each word appears in the idiom. Recurrent networks are distinguished because every network layer uses the same parameters [23]. Recurrent neural networks share a similar weight parameter inside each layer of the network, unlike feedforward networks, which have distinct weights across each node. Nonetheless, to support reinforcement learning, these weights are still modified using the techniques of backpropagation and gradient descent. RNNs use the backpropagation through time (BPTT) method, which differs compares conventional backpropagation since that was tailored to linear data to find the gradients. In classical backpropagation, the model learns by calculating errors from the outcome layer to the input layer. These computations enable us to accurately alter and fit the model's parameters in contrast to feedforward networks.

Using binary classifiers for the sentiments from the feature vectors for each review. Few classifier models are applied from the recurrent neural networks, which are

- Bidirectional – LSTM
- GRU

The selection of algorithms is crucial to achieving accurate and meaningful results. Bi-LSTM and GRU are chosen for modelling sequential data and have shown significant effectiveness in Text Sentiment Analysis tasks. Bi-LSTM enables the model to leverage context from preceding and succeeding words, providing a richer text representation. Both Bi-LSTM and GRU offer advantages in capturing long-term dependencies and contextual information and mitigating vanishing gradients.

C. *LSTM*

An extended short-term memory network (LSTM) is a type of complex RNN or sequential network that can retain information over time, addressing the vanishing gradient problem of traditional RNNs. LSTMs are often used for tasks requiring long-term memory. RNNs retain their prior knowledge and use it while processing the input at hand.

Long-term dependencies can be a challenge for RNNs due to diminishing gradients. LSTMs can capture and leverage the contextual information present in the text, including syntactic and semantic structures, which is crucial for accurately interpreting sentiment. LSTMs can effectively retain and propagate information over longer sequences, allowing them to grasp sentiment-related nuances across multiple words or phrases. However, LSTMs are designed to address this issue and can retain information over more extended periods. LSTM functions on a high level, very similar to an RNN cell. The LSTM network's internal operation is seen below. There are four separate gates and each serving a different function that make up an extended short-term memory network. These gates are as follows,

- Forget gate

  The input and previous output are combined at the forget gate to produce a fraction between 0 and 1, indicating how much of the initial state must remain. The previous state is multiplied by this output after that progress. It should be noted that an activation output of 1.0 indicates "remember everything," while a value of 0.0 indicates "forget everything." The "remember gate" could be a more fitting moniker when seen from a different angle than "forget gate".

- Input gate

The input gate of an LSTM utilizes frequency signals like the forget gate, but its function is to determine which new information should be permitted to enter the state of LSTM. The new values that must be added to the initial state are produced by multiplying the input gate's output with the tan h block (a fraction between 0 and 1). Next, the previous state is combined with this gated vector to create the present state.

- Input modulation gate

It is frequently regarded as a component of the input gate, and much of the literature on LSTMs assumes it is located inside the input gate without even mentioning it. By giving the information non-linearity and making it zero-mean, the purpose of the output gate is to regulate the flow of information from the Internal State Cell to the hidden state output. It uses the current input and previous hidden state to determine which information should be outputted. Due to the faster convergence of zero-mean input, this is done to shorten learning time. While the activities of this gate are less significant than those of the others and are sometimes seen as a finesse-providing notion, it is advisable to include it in the LSTM unit's construction.

- Output gate

The previous state and input are gated at the output gate to produce another scaling fraction, combined with the tanh block's output to get the current state. The scaling fraction refers to the component that controls the flow of information from the previous state and input to the current state. The scaling fraction weighs the importance of the previous state information with the new input. A value of 0 would mean that the previous state is entirely ignored, while a value of 1 means that the previous state's information is fully retained. Including the scaling fraction in the LSTM architecture allows the network to control the flow of information and selectively retain or discard information from previous states based on the current input and the learning task at hand. The result is then distributed. The LSTM block receives input from both the output and the state.

D. GRU

A type of recurrent neural network called Gated Recurrent Units (GRUs) is a more straightforward version of the Long Short-Term Memory architecture. GRUs may analyze sequential data similarly to LSTMs by selectively storing and losing information over time. GRUs, on the other hand, do this by employing a more straightforward architecture with fewer calculations and gates, which makes them quicker to calculate and more accessible to train than LSTMs. The use of two types of gates, a reset gate and an update gate, to manage the information flow through the model is the main innovation of GRUs, in contrast to the reset gate, which decides how many of the previous state to forget and how many of the new input to remember.

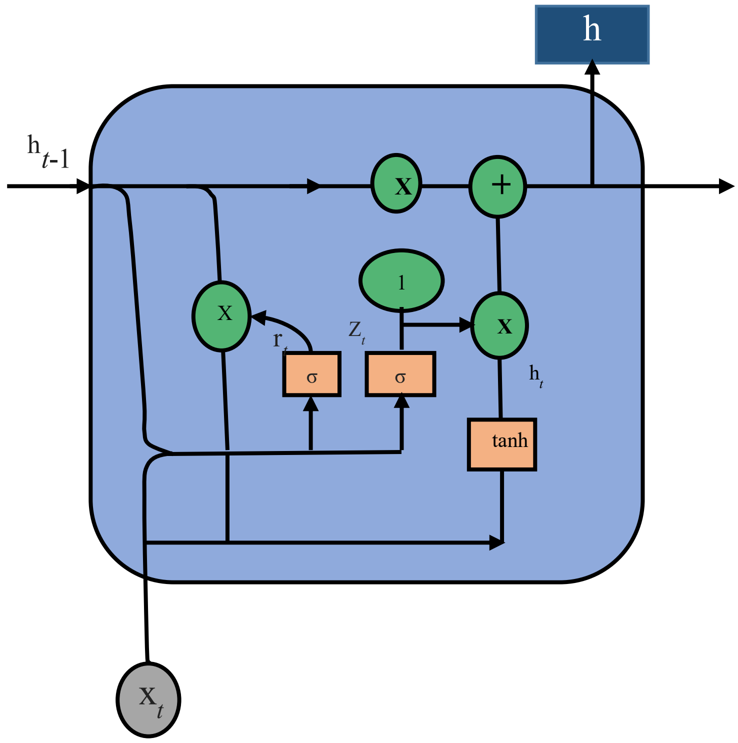

Fig 1. Architecture of GRU

The update gate chooses how many previous states to keep and how many new inputs to add. A GRU's update gate is a sigmoid function that accepts the previous hidden state and the current input as inputs and outputs a value between 0 and 1 that specifies how much of the initial state to maintain and how much of the new input to add. Some 0 indicates that the initial state has been entirely forgotten, whereas a value of 1 indicates that it has been entirely kept. The reset gate in a GRU is likewise a sigmoid function that accepts the concatenation of the previous hidden state and the current input as input and returns a number between 0 and 1 that specifies how much of the previous state to forget and how much of the new input to remember. Several 0 indicates that the initial state is entirely preserved, whereas a value of 1 indicates that it is entirely forgotten in Fig 1.

Using the update and reset gates, the GRU's current hidden state is calculated by adding its initial hidden state to its current input, in contrast to the reset gate, which decides how much of the previous hidden state to forget and how much of the new input to remember, the update gate chooses how much of the previous hidden state to keep and how much of the new input to add. As with an LSTM, this enables the model to retain and forget information over time selectively. The simplicity of its design, which makes GRUs easier to train and faster to calculate, is one of their main advantages over LSTMs. GRUs are less prone to overfitting and can be trained on fewer datasets than LSTMs since it contains fewer parameters and calculations than the latter. Moreover, GRUs are resource-constrained devices like mobile phones and Internet of Things (IoT) devices well-suited for deployment because of their computational efficiency.

E. Bidirectional LSTM

A Neural Network design called Bi-directional Long Short-Term Memory (Bi-LSTM) enhances the capabilities of the standard LSTM model by processing the input sequence forward and backwards. In addition to other sequential data tasks like speech recognition and activity recognition, Bi-LSTMs have been extensively employed in natural language processing tasks, including machine translation, named entity identification, and sentiment analysis. LSTMs can process sequential input by selectively storing and forgetting information over time. The input sequence is only ever processed in one way, either from right to left or from left to right, in a conventional LSTM. A bi-LSTM, on the other hand, processes the input sequence concurrently in both directions using two different LSTM layers. At each time step, the outputs from these two layers are concatenated to provide a final output that includes data from the input sequence's past and future in Fig 2.

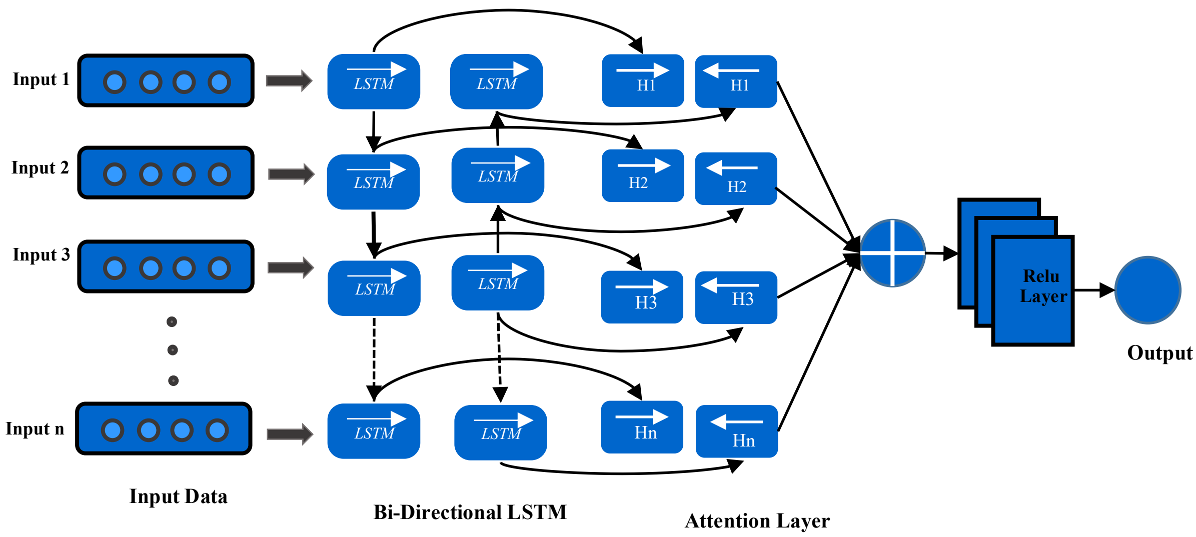

Fig 2. Architecture of Bi-LSTM

The main benefit of a Bi-LSTM over a regular LSTM is that it enables the model to capture the context of past, present, and future time steps and the context of the current time step. This is especially helpful in activities where correct forecasts depend on the past and future context. For instance, it's crucial to consider the surrounding words in a phrase when identifying identified entities. A Bi-LSTM can produce more precise predictions because it can understand the context of both the words that come before and after them in the phrase. A bi-LSTM model's architecture consists of two LSTM layers; one processes the input sequence from right to left and the other from left to right. Information flows into and out of the cell state is regulated by a series of gates in each LSTM layer. The model can selectively store and forget information over time thanks to these gates, which contain the input, forget, and output gates. The two LSTM layers' outputs are combined at each time step to produce a final output incorporating knowledge from the input sequence's past and future after passing this final output via one or more wholly linked layers to achieve the final classification progress.

The Rectified Linear Unit (ReLU) function is widely used in deep learning models as an activation function, and it is also commonly applied in Bi-LSTM models to enhance accuracy. The ReLU function is utilized on the output of the LSTM layer in a Bi-LSTM to enhance the model's non-linearity. When the input value is positive, the ReLU function returns it; otherwise, it returns zero. This non-linear activation function is crucial since it enables the model to learn complex relationships between the input and output. Incorporating the ReLU function in the Bi-LSTM's LSTM layer output allows the model to recognize intricate dependencies in the data and improve its prediction accuracy. Furthermore, the ReLU function is computationally efficient and easy to implement, which makes it a widely accepted choice in deep learning models. In conclusion, using the ReLU activation function in a Bi-LSTM can significantly enhance the model's accuracy and help capture complex relationships in the data.

V. EXPERIMENTAL RESULTS

In the research, they came up with many NLP categorization methods. The investigations were divided into two primary categories. In the first section, apply the bag of words and skip-gram word2vec models on the dataset to describe words quantitatively. The following section would be the binary classification job that was then carried out using a variety of classifiers, including Bidirectional LSTM and GRU. Fig 3 and 4 above explain the accuracy and loss comparison of the bidirectional LSTM model.

Fig 3. Accuracy of Bi-LSTM model

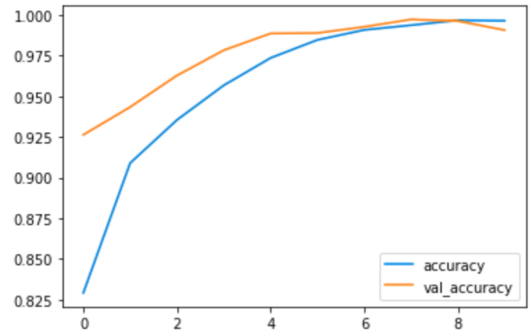

Fig 4. Loss of Bi-LSTM model

The loss is generated by plotting the loss function values throughout training. The loss function represents how well the model performs regarding its ability to make accurate predictions. The model iteratively adjusts its internal parameters during training to minimize this loss function.

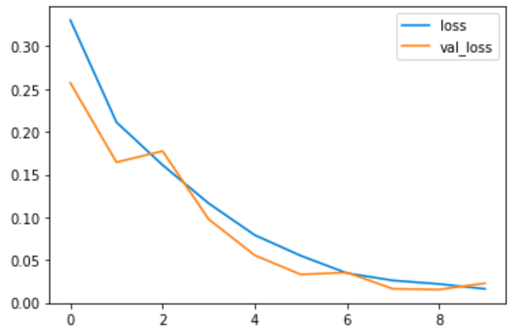

Fig 5. Accuracy of Gated Recurrent Unit model

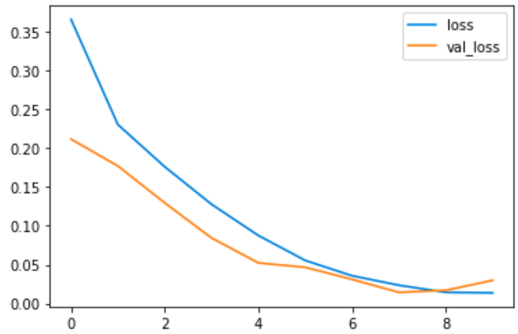

Fig 6. Loss of Gated recurrent Unit model

The above-shown Fig 5 and 6 explains the accuracy and loss comparison of the Gated Recurrent Unit model. Both the Bi-LSTM and GRU are deep learning techniques that bring high accuracy in the implementation. GRU achieved 98.24% of accuracy and bi-LSTM achieved 98.65% of accuracy. GRU has a better performance compared to bi-LSTM but when the data is too big GRU is unworthy. Bi-LSTM can handle huge data even though it takes much time to progress but the accuracy of the model always will be high.

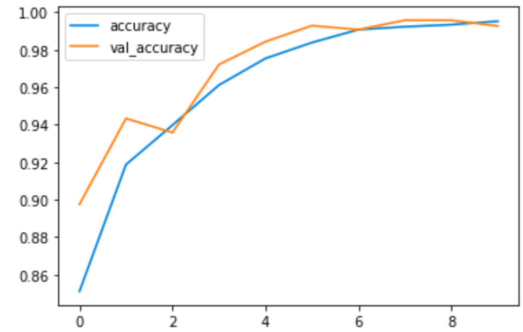

TABLE III. RESULT TABLE

Training data	Validation Data	Model used	Class	Accuracy
25000	25000	Bi-LSTM	ID, Sentiment, Review	98.45%
25000	25000	GRU	ID, Sentiment, Review	98.24%
25000	25000	Bi-LSTM with ReLu function	ID, Sentiment, Review	99.65%

The training data, validation data, and attributes used in the respective models and their accuracy are mentioned in Table 3. The Bi-LSTM works better than GRU in terms of high accuracy. Table 4 explains the number of epochs implemented in both models and their rising accuracy value is mentioned. This clearly shows that the accuracy gradually increases when the number of epochs increases. The limitations of the work might be finding the dataset with high quality and diversity availability. While these models may achieve high accuracy, understanding the underlying reasons for their predictions can be challenging.

TABLE IV. ACCURACY TABLE

No of epochs	1	2	3	4	5	6	7	8	9	10
Bi-LSTM	85.1	90.8	92.9	95.1	96.5	97.3	98.1	98.2	98.3	98.45
GRU	81.9	84.7	87.3	89.8	92.5	94.6	96.3	97.4	98.1	98.24
Bi-LSTM with ReLU	85.1	91.8	93.9	96.1	97.5	98.3	99.1	99.2	99.3	99.65

VI. DISCUSSION AND CONCLUSION

A model for sentiment analysis using Bi-LSTM and GRU attention was presented. The suggested model outperforms the current models and its evaluation trials utilizing data from IMDB movie reviews. The proposed work helps investigate Bi-LSTM and GRU's effectiveness in sentiment analysis. The suggested model's accuracy improved as the amount of training data and the number of training epochs increased. This offers an alternate solution to the issues with traditional models' long-term dependencies and data loss that arises as training data volume grows. Rather than improving accuracy, text classification research has focused on extracting precise semantics and characteristics from specific domains and sectors needing specialist expertise (e.g., medical, engineering, and emotional).

VII. FUTURE SCOPE

Text categorization has recently been used in specialist domains, including clinical, legal, medical, and commercial marketing or marketing business [24], sentiment analysis and clinical settings. Much more is needed in these disciplines than just sentiment categorization. As part of the study, the intention was to investigate more comprehensive applications of sentiment categorization to additional analytic disciplines. Additional cutting-edge strategies, such as transfer learning, are used to gather sufficient training data to apply to the model. To improve multi-class prediction and also want to increase the classification labels. The currently suggested model combines some existing models and has obvious flaws. In future works, the main focus will be developing new methods or different architectures to address the issue raised. Increasing the diversity and size of the training data can enhance model performance. An ensemble model can be developed, which may increase the model's performance.

VIII. REFERENCES

[1] Kottursamy, Kottilingam. "A review on finding efficient approach to detect customer emotion analysis using deep learning analysis." Journal of Trends in Computer Science and Smart Technology 3, no. 2 (2021): 95-113.

[2] Loureiro, Sandra Maria Correia, João Guerreiro, Sara Eloy, Daniela Langaro, and Padma Panchapakesan. "Understanding the use of Virtual Reality in Marketing: A text mining-based review." Journal of Business Research 100 (2019): 514-530.

[3] M. Balaji and N. Yuvaraj., “Intelligent Chatbot Model to Enhance the Emotion Detection in social media using Bi-directional Recurrent Neural Network”. Journal of Physics: Conference Series, International Conference on Physics and Photonics Processes in Nano Sciences, vol. 1362, pp. 20–22, 2019.

[4] Lei Jiang, Yuan Li, Jing Liao, Ziwei Zou, Caoqing Jiang, Research on non-dependent aspect-level sentiment analysis, Knowledge-Based Systems, Volume 266, 2023.

[5] Zilinghan Li, Xiwei Wang, Zhenning Zhang, Volodymyr Kindratenko, ViCTer: A semi-supervised video character tracker, Machine Learning with Applications, Volume 12, 2023.

[6] Duarte, J.M., Berton, L. A review of semi-supervised learning for text classification. Artif Intell Rev (2023).

[7] Pouransari, Hadi, and Saman Ghili. "Deep learning for sentiment analysis of movie reviews." CS224N Proj (2014): 1-8.

[8] Melamud, Oren, Jacob Goldberger, and Ido Dagan. "context2vec: Learning generic context embedding with bidirectional lstm." In Proceedings of the 20th SIGNLL conference on computational natural language learning, pp. 51-61. 2016.

[9] Ceraj, Tin, Ivan Kliman, and Mateo Kutnjak. "Redefining cancer treatment: comparison of Word2vec embeddings using deep Bi-LSTM classification model." Text Analysis and Retrieval 2019 Course Project Reports 124, no. 24 (2019).

[10] Xiao, Lizhong, Guangzhong Wang, and Yang Zuo. "Research on patent text classification based on word2vec and LSTM." In 2018 11th International Symposium on Computational Intelligence and Design (ISCID), vol. 1, pp. 71-74. IEEE, 2018.

[11] Rehman, Anwar Ur, Ahmad Kamran Malik, Basit Raza, and Waqar Ali. "A hybrid CNN-LSTM model for improving the accuracy of movie reviews sentiment analysis." Multimedia Tools and Applications 78 (2019): 26597-26613.

[12] Küpper, Axel, Younghee Park, Peter Ruppel, Stefan Schulte, and Jie Xu. Proceedings of the 2019 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPCON 2019). IEEE, 2019.

[13] Wang, Jin, Liang-Chih Yu, K. Robert Lai, and Xuejie Zhang. "Tree-structured regional CNN-LSTM model for dimensional sentiment analysis." IEEE/ACM Transactions on Audio, Speech, and Language Processing 28 (2019): 581-591.

[14] She, Xiangyang, and Di Zhang. "Text classification based on hybrid CNN-LSTM hybrid model." In 2018 11th International symposium on computational intelligence and design (ISCID), vol. 2, pp. 185-189. IEEE, 2018.

[15] Salur, Mehmet Umut, and Ilhan Aydin. "A novel hybrid deep learning model for sentiment classification." IEEE Access 8 (2020): 58080-58093.

[16] Zhang, Jiarui, Yingxiang Li, Juan Tian, and Tongyan Li. "LSTM-CNN hybrid model for text classification." In 2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), pp. 1675-1680. IEEE, 2018.

[17] Dong, Yanru, Peiyu Liu, Zhenfang Zhu, Qicai Wang, and Qiuyue Zhang. "A fusion model-based label embedding and self-interaction attention for text classification." IEEE Access 8 (2019): 30548-30559.

[18] Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).

[19] Jang, Beakcheol, Myeonghwi Kim, Gaspard Harerimana, Sang-ug Kang, and Jong Wook Kim. "Bi-LSTM model to increase accuracy in text classification: Combining Word2vec CNN and attention mechanism." Applied Sciences 10, no. 17 (2020): 5841.

[20] Amulya, K., S. B. Swathi, P. Kamakshi, and Y. Bhavani. "Sentiment Analysis on IMDB Movie Reviews using Machine Learning and Deep Learning Algorithms." In 2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT), pp. 814-819. IEEE, 2022.

[21] Ramadhan, Nur Ghaniaviyanto, and Teguh Ikhlas Ramadhan. "Analysis sentiment based on IMDB aspects from movie reviews using SVM." Sinkron: jurnal dan penelitian teknik informatika 7, no. 1 (2022): 39-45.

[22] Sabba, Sara, Nahla Chekired, Hana Katab, Nassira Chekkai, and Mohammed Chalbi. "Sentiment Analysis for IMDb Reviews Using Deep Learning Classifier." In 2022 7th International Conference on Image and Signal Processing and their Applications (ISPA), pp. 1-6. IEEE, 2022.

[23] Maas, Andrew, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. "Learning word vectors for sentiment analysis." In Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies, pp. 142-150. 2011.

[24] Mouthami Kuppusamy, Anandamurugan Selvaraj, “A novel hybrid deep learning model for aspect based sentiment analysis”, Concurrency and Computation: Practice and Experience, no 35, 1-13, 2022.