Weighted Ensemble LSTM Model with Word Embedding Attention for E-Commerce Product Recommendation

Prashant Sharma, and Vijaya Ravindra Sagvekar

Original scientific article

***Abstract*—Nowadays, the proliferation of social media and e-commerce platforms is largely due to the development of internet technology. Additionally, consumers are used to the idea of using these platforms to share their thoughts and feelings with others through text or multimedia data. However, it is difficult to identify the best categorization methods for this type of data. Furthermore, users are seen to have difficulty understanding aspect-based feelings conveyed by other users, and the currently existing models' accuracies are not up to par. Deep learning models used for sentiment analysis (SA) provide improved performance by finding out the actual emotions in the presented data. The aim of this research is to develop a weighted ensemble with Long Short-Term Memory (LSTM), and a specialised deep learning model using unique word embedding approaches to enhance its performance in sentiment analysis. The words with a strong connection to a particular class are given more weight by the Word Embedding Attention (WEA) technique. The weighted ensemble with LSTM yields superior outcomes because of its excellent generalization capabilities. By integrating the advantages of several models and mitigating the effects of each model's shortcomings, ensemble voting raises the prediction accuracy. By lessening the influence of outliers or errors in individual model categorization, ensemble voting increases the robustness of categorization. This LSTM weighted ensemble achieves 99.82 % accuracy, 99.4% precision, 99.02% f-score, and 99.7% recall in sentiment analysis which is much higher when compared to the outcomes of conventional methods.**

***Index terms*—Deep learning, Long Short-Term Memory, Sentiment analysis, Weighted Ensemble, Word embedding attention.**

I. INTRODUCTION

Sentiment analysis, otherwise known as opinion mining is the process of identifying, extracting, and classifying subjective data from unorganised language by employing text analysis and computerised linguistic methodologies [1]. In this regard, a variety of social media resources, including blogs, reviews, articles, and tweets are processed to glean opinions from the public regarding a specific entity or circumstance [2].

Manuscript received September 14, 2023; revised October 9, 2023. Date of publication December 12, 2023. Date of current version December 12, 2023. The associate editor prof. Ivo Stančić has been coordinating the review of this manuscript and approved it for publication.

P. Sharma is with the Department of Computer Science and Engineering, Mandsaur University, Mandsaur (MP), India (e-mail: rk.prashantsharma@gmail.com).

V. R. Sagvekar is with the Information Technology Department, Vasantdada Patil Pratishthan's College of Engineering & Visual Arts, Sion-Chunabhatti, India (e-mail: vijayabijoor@pvppcoe.ac.in, vijayasagvekar@gmail.com).

Digital Object Identifier (DOI): 10.24138/jcomss-2023-0126

The process of sentiment analysis examines the ideas, views, attitudes, feelings, and impressions that people share on various social media platforms [3, 4]. Aspect-Based Sentiment Analysis (ABSA) extends a step to further examine the users' sentiments and views about an entity or an occasion in a specific, detailed way [5]. The sentiment analysis determines whether a text's polarity (positive or negative viewpoint) is present in any given clause, paragraph, or sentence [6]. It seeks to look into how people feel about many things, including society, topics, events, and objects, as expressed through text evaluations or reviews on social media (such as forums, social networking sites, blogs, etc.) [7]. One well-known online retailer that allows for open product evaluation and reviews by users is Amazon. Users' decision-making, which may include purchasing a phone, camera or other product, or writing a movie review, or investing, is aided by evaluating these reviews and classifying them as positive or negative. This activity significantly impacts the users' daily lives. A number of data mining approaches are available to perform sentiment analysis, which are repeatedly improvised to scale up their accuracies. The earlier works took into account only the user and product information as distinct characteristics that are individually incorporated into text representation [8]. Almost everyone is now able to communicate their thoughts and opinions online through the social media's rapid expansion. Therefore, sentiment analysis is essential for gaining a proper understanding of what consumers or reviewers think [9].

It is difficult to identify, track, and filter data available on social media applications in order to analyze sentiments. Chaotic information arises from a variety of sources including language differences, abundance of online sites and social media, and also a vast knowledge of personal experiences. In this case, it is necessary to analyse the sentiments of the data by using relevant tools [10]. Majority of the prior approaches consider user-marked ratings or polarity as the criteria, while machine learning algorithms are used to train the sentiment classifiers with text characteristics in order to achieve a good benchmarks set in the document-level sentiment classification [11]. Progress is made through the incorporation of deep learning in many NLP tasks such as text sentiment analysis. For predicting emotions in text, several academicians concentrate on creating neural network-based sentiment analysis algorithms [12]. Sentence-level SA which takes responsibility for both the subjectivity and objectivity of a sentence is a developing area for text mining in feature learning [13]. Furtherly, SA tasks fundamentally involve four steps: preprocessing, feature extraction, classification, and result interpretation. These procedures are applied into a variety of data domains such as movie reviews, airline reviews for election opinion prediction, Amazon reviews [14], etc. Among all the procedures described above, feature extraction is crucial for increasing the classification accuracy [15]. Understanding user behaviour is becoming increasingly important as social media data grows at a rapid pace due to user contributions, particularly in light of the recent topics [16]. The scope of the studied dataset in this research are the thoughts expressed in the form of posts about the epidemic. It is difficult to find the ideal classification method for this type of data. In this setting, deep learning models for sentiment analysis have shown the potential to improve performance of current feature-based methods and also enabled extensive representation capabilities [17]. However, there is currently not a lot of research regarding the use of ensemble learning in social media applications for sentiment analysis. As a result, the objective of this research project is to contribute to the collection of literature through the analysis of opinions generated from social media applications' data, by using the ensemble approach which integrates many deep learning models [18]. Since it is possible to prevent unrelated errors produced by a single classifier, the usage of an ensemble of classifiers is proven to be a very effective technique to increase the classifying efficiency. With the hope of assisting in E-Commerce Product Recommendations, this research improves upon the weighted voting ensemble learning technique. It aims to create an efficient feature selection approach for sentiment analysis. The primary goal of this research is to attain dimensionality reduction in addition to generating high-quality outcomes from the text classification tasks. The main contributions are mentioned below:

- The user interests and feelings examined across different locations are taken into consideration to treat the varying viewpoints under different geographic regions.
- The implemented weighted ensemble with LSTM is proved to be beneficial in dealing with long-term reliance issues through utilizing the self-feedback system in the hidden layer.
- Because client interests fluctuate from time to time, current trends are also addressed in sentiment analysis.
- Feature extraction is performed utilizing bag of words-based word embedding techniques to save time when searching for underutilized features.

The other sections of this study are organized as follows: section II presents the Literature Review, and section III describes the methodology's performance. Section IV presents the findings and discussions while section V provides the research paper's conclusion.

II. Literature Survey

Categorizing the sentiment polarities in social network data utilizing convolutional neural network (CNN) was proven by Meena et al. [21]. The classification of the many reviews provided by members of different ethnic groups was the main objective of this study. Real-time data taken from a social media network was used for the testing. Sentiment polarities which are positive, negative and neutral were employed to classify the opinions. Another purpose was for commercial enterprises and their clientele to be able to forge strong bonds by implementing the suggested structure. This paper also covered the various uses of sentiment analysis methods, especially in the fields of security and social networking.

A hybrid deep learning strategy based on Convolutional Neural Network Long Short-Term Memory (CNN-LSTM) was presented by Mohbey et al. [22] for sentiment analysis of monkeypox tweets. The main objective was to find out how the public feels about the then current monkeypox outbreak so as to help policymakers have a better understanding of how the public perceives the disease. Several investigations were carried out using an accessible dataset comprising tweets on the monkeypox. The tweets went through a number of one-hot encoding, global vectorization, and pre-processing steps. The investigation's findings helped raise awareness of the monkeypox virus among the general public.

Sivakumar and Uyyala [23] implemented a Long Short-Term Memory With Fuzzy Logic (LSTM-FL) based on an intelligent system to categorize statements of user review into four categories as extremely positive, positive, extremely negative, and negative. The implementation also examined user feedback in light of geographical location and current trends. The learning rate of the LSTM network was adjusted using the default Adam optimizer. By using the Adam optimizer, the implemented model's accuracy was increased. Even though the sequential dependencies in data were exceptionally well captured by the LSTM models, comprehending the contextual information was proved to be troublesome.

Shobana and Murali [24] implemented an APSO-LSTM Skip-gram architecture that was utilized to obtain greater contextual information of words, and for semantic feature extraction. To understand the input of intricate textual patterns, the implemented approach made use of long short-term memory. The words were mentioned in lower-dimensional space and the depiction was highly accurate after using a word2vec architectural model based on the implemented skip-gram. However, it was difficult to understand the LSTM's functions and the spot biases or inaccuracies in sentiment predictions as these were complicated models with a lot of parameters.

Basiri et al. [25] implemented an Attention-based Bidirectional CNN-RNN Deep Model (ABCDM) for sentiment analysis. Employing two independent bidirectional LSTM and GRU layers, the ABCDM collected contexts from the past as well as the future by considering the temporal flow of information across all orientations. The dataset's experimental findings showed that ABCDM produced advanced classification outcomes for both extensive reviews and short tweets. However, this model became computationally complex and resource-intensive which slowed down the training and deployment, as opposed to the simpler models.

Zhao et al. [26] implemented a cross-domain sentiment classification (CDSC) method through parameter transferring and attention-sharing mechanism (PTASM). The developed structure contained the target domain network (TDN) and the software domain network (SDN). The model enhanced the discriminative power by capturing shared and domain-specific input, along with improving the sentiment classification accuracy. This method proved to be highly efficient in cross-domain sentiment transfer. However, the transferred parameters did not reflect the unique properties of the target domain, thereby resulting in a loss of classification performance.

Sazzed and Jayarathna [27] implemented SsentiA, a hybrid methodology of self-supervision which used no labelled data and operated completely unsupervised. SSentiA was applied to both document-level and sentence-level classification of sentiments. The implemented SSentiA employed a self-supervised approach for accurately and reliably classifying the captured sentiments. However, SSentiA had trouble achieving high accuracy or robustness in sentiment classification because the availability of labelled data was limited, or notably different from the unlabelled data.

Dadhich and Thankachan [28] implemented a Product Comment Summarizer and Analyzer (PCA) system design that was speedy, generic, and robust, and categorized the online English comments gathered from shopping websites of Flipkart and Amazon. Five different classification techniques were used for the supervised learning. With the entire dataset, the PCSA system carried out the classification of comments very effectively and with high accuracy. Nonetheless, the sentiment analysis relied on the imbalanced training data which led to skewed or inaccurate results.

Atandoh et al. [29] suggested BERT-MultiLayered Convolutional Neural Network (B-MLCNN) which was as an integrated deep learning paradigm that was computationally feasible. The B-MLCNN categorized the available emotions and treated the entire collection of textual reviews as a single document. The feature vector representation was handled by the BERT pre-trained language model which also recorded the global characteristics. Moreover, feature extraction was handled by the MLCNN with different kernel dimensions. The better text semantic and syntactic feature extraction, and classification performance were thus accomplished by the used approach. Nonetheless, additional training was frequently necessary for the correct generalization of massive volumes of the tagged data.

An effective LSTM-based sentiment analysis of e-commerce reviews was provided by Gondhi et al. [30]. This study suggested the classification of a significant proportion of Amazon reviews using LSTM networks. This deep learning method was quick and produced better outcomes even for a high volume of evaluations. The work estimated word representations in the vector space efficiently by using word2vec embedding. Compared to conventional representation techniques like bag of words or one-part encoding, the word2vec yielded superior results. This work primarily focused on two areas: the LSTM network for review classification, and the effective mapping of emotion words into vector space using the word2vec model. Unfortunately, there were issues with vanishing gradients, overfitting, and slow convergence time with this model.

The word embedding attention and balanced cross techniques used for sentiment analysis have certain limitations. The major drawbacks are their dependences on imbalanced training data. When the training data contains a significant imbalance between the number of positive and negative sentiment samples, it leads to skewed or inaccurate results during sentiment classification. This imbalance causes the model to favor the majority class, leading to reduced performance in correctly identifying minority sentiments.

III. Methodology

The implemented deep-learning language framework of weighted ensemble LSTM for sentiment analysis which consists of an Amazon dataset, data preprocessing, syntactic and semantic labels, word embedding layer, proposed language mode based on deep learning, analyzing sentiments taken from google, Sentiment API of Microsoft Azure and IBM Watson, sentiment polarity class probability, ensemble-based Weighted voting, final sentiment analysis for Word Embedding Attention and Balanced Cross Entropy technique, is represented in Figure 1.

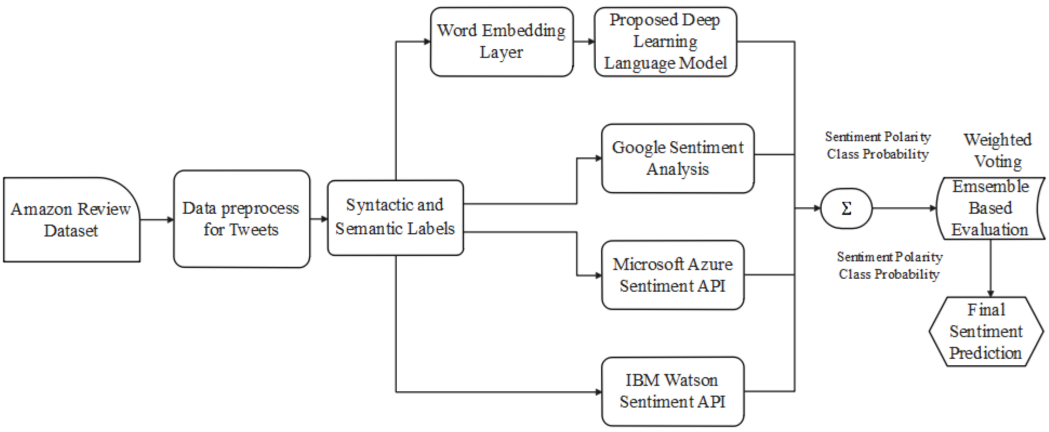

Fig. 1. The implemented weighted ensemble LSTM for sentiment analysis

A. Dataset

The Amazon review dataset is used to acquire behaviour-based data. The user's behaviour file and the product list file are the main records from the dataset that were gathered. The user's behavioural file contains information about the user's background comprising the dwell time, clickstream, and a browsing module. The User ID, product name (such as books, electronics, clothes, etc.), purchases, session ID, product ID, number of clicks, time, and category are the data types that were acquired from the e-commerce website. The acquired data is frequently scheduled within the user requests based on the user data. 50,000+ Amazon user reviews and star ratings are included in Amazon dataset. The 50,000+ reviews are broken down into 11,806+ neural reviews, 15,567+ bad reviews, and 22,627+ good reviews [31].

B. Data Preprocessing

The data provided by social media users includes content besides alphabetic characters like usernames, words, punctuation, web links, URLs, and graphical icons. These contents do not aid in any of the sentiment analysis procedures. For instance, none of the algorithms support usernames during the categorization of tweets as negative or positive. Therefore it is appropriate to remove such content, frequently referred to as noise, for improving the effectiveness of algorithms that perform classification.

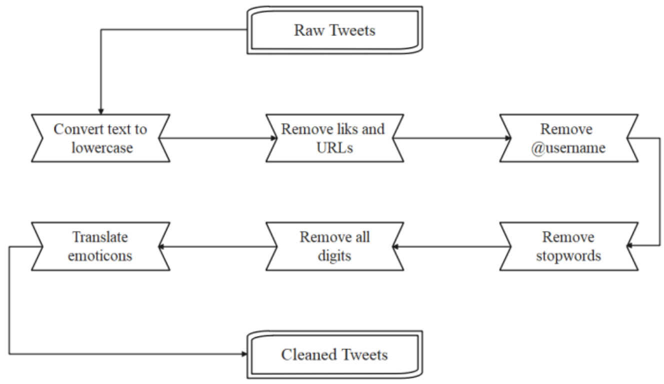

Fig. 2. Data preprocessing pipeline for dataset

The data processing steps employed in this research are depicted in Figure 2. The text of all characters is changed to lowercase in the pipeline's initial stage. All web links, URLs, and usernames are deleted because these do not contribute to the text's emotional or sentimental value. Further on, numbers, punctuation, and undefined characters are also eliminated from this pipeline. The next step in data processing is converting emotions and graphical indicators into negative or positive polarity. This translation is then utilized to give each tweet a class name.

C. Syntactic and Semantic Labels

Text Blob is a sentiment analyzer based on Lexicon. It includes particular predefined rules or a word and dictionary of weight with some scores that allow for assessment of statement polarity. The "Rule-based sentiment analyzers" is another name for the Lexicon-based sentiment analysis tool. It is a Python 2 and 3 library used to process textual data. It offers an easier API for dividing into Natural Language Processing (NLP) activities like analyzing and classifying sentiments, and for the extraction of noun phrases that define positive, neutral, and negative. Text Blob has a unique feature that handles modifiers, also known as intensifiers, which enhance the meaning of the text based on their pattern. When a modifier word is present, Text Blob ignores the polarity and opinion instead of relying exclusively on the intensity to determine the text sentiment.

D. Word Embedding Layer

As there are currently no contextual relationships between words, the model of bag-of-words in general appears extremely and highly dimensional. By identifying suffixes and prefixes in training data, the unseen or rare words are recognized and the contextual relationships between the words are learnt. This is extremely helpful when dealing with epidemic information that uses new terms and words. In this system, each word's representation includes the word itself, as well as a collection of characters of the n-gram. For instance, the character n-grams for the word "matter" with n=3 are represented as <ma, mat, att, tte, ter, er>. In this case, brackets are inserted as symbolic boundaries to distinguish the n-grams of a word from its original form. If any components from the vocabulary include the words such as "mat", it is rendered as <mat>. This situation helps in the preservation of shorter words, meaning that it is possible that they are present in n-grams of longer ones. Additionally, it enables the prefixes and suffixes to capture their underlying meaning.

E. Proposed Deep Learning Model LSTM

The LSTM network receives the feature set in series after each word is represented by its matching feature vector, following the utilization of the model for word embedding. The long-range relations that are visible in the input are captured by the LSTM model as it stores past data. In sequence modeling applications like text classification, sentiment analysis, time series prediction, etc., the LSTM consistently outperforms its competitors in the field to a great extent. The forget gate, input gate, and output gate are three crucial components of the LSTM model. The forget gate chooses to delete or ignore irrelevant data from the most recent input data as well as from the prior cell state. To return the values between [0,1], a sigmoid function is employed during training. A value that is close to 0 indicates that updating is not as necessary. Then it is not a crucial requirement to remember the information. Thus, the input gate serves as a filter, determining whether the data should be worth remembering to be updated into the next state. The information of the output in the following cell state is decided by the output gate. Figure 3 shows the LSTM network's fundamental design.

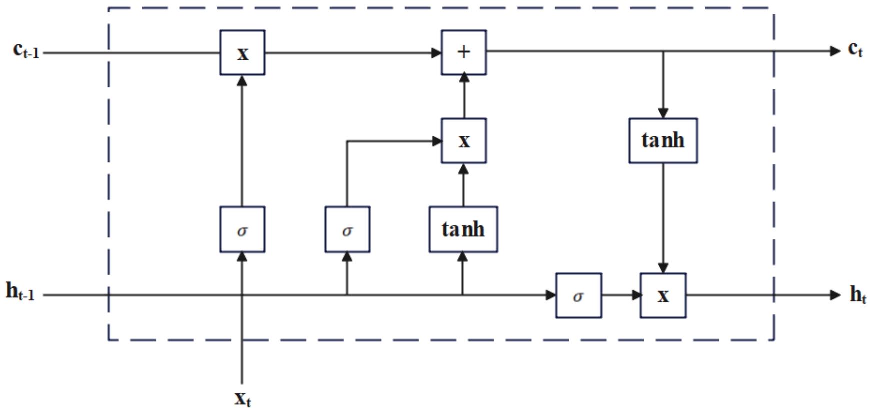

Fig. 3. The fundamental design of the LSTM Cell

Hidden state and cell state are the two states that form the LSTM. The LSTM determines which data must be derived from the cell state at a specific time t. A layer of sigmoid function σ, known as the forget gate makes the choice. The function outputs a number between [0,1] using the inputs x_t and h_{t-1} which is the output from the previous hidden layer. In the equation (1) that follows, 0 in this instance stands for "completely taken away", and 1 for "completely keeping in". The LSTM [32] is proved to be beneficial in dealing with long-term reliance issues, and has the capability to generalize and learn from the input information. The LSTM assists in decreasing the overfitting, that is especially significant in sentiment analysis to establish the unseen text generalization model. For keeping the data in the LSTM model, the memory cell primarily consists of three gates: an input gate, a forgetting gate, and an output gate. This arrangement helps to address the issues with long-term features.

The output is stated as h_t while the LSTM cell output is stated as h_{t-1}. In LSTM, bias term is signified as b_c, the memory cell is stated as $\tilde{c_t}$, and the weight matrix is stated as W_c, which is explained in equation (1),

$$\tilde{c_t} = tanh(W_c.[h_{t-1}, x_t] + b_c) \quad (1)$$

The input gate is signified as i_t, the current input data manages a state value with the help of i_t, b_i signifies a bias, weight matrix is signified as W_i, and the sigmoid function is stated as σ. The input gate i_t is mathematically defined in the below equation (2),

$$i_t = \sigma\left(W_i.\left[h_{t-1}, x_t\right] + b_i\right) \quad (2)$$

f_t signifies the forget gate value that assesses the memory based on the predicted data. It is upgraded by managing f_t which is mathematically defined in equation (3),

$$f_t = \sigma\left(W_i.\left[h_{t-1}, x_t\right] + b_f\right) \quad (3)$$

Memory cell is signified as c_t while the value of unit state is signified as c_{t-1} which is expressed in equation (4),

$$c_t = f_t * c_{t-1} + i_t * \tilde{c_t}. \quad (4)$$

In the equation (5) '*' is recognized as dot product. o_t is referred as the output gate which is computed by using the memory cell state and is regulated by the output gate. o_t is stated in the equation below (5),

$$o_t = \sigma\left(W_0.\left[h_{t-1}, x_t\right] + b_0\right) \quad (5)$$

The output h_t attained from LSTM is designed by using the equation (6),

$$h_t = o_t * tanh\left(c_t\right) \quad (6)$$

The hidden forward layer's hidden unit function $\vec{h}$ at each interval step t is calculated by x_t which is input current step and h_{t-1} which is the previous hidden state. Hidden unit function $\overleftarrow{h}$ of a hidden backward layer is calculated by x_t and the future hidden state $\overleftarrow{h}_{t+1}$. Depictions of forward and backward circumstances are created through $\vec{h}_t$ and $\overleftarrow{h}_t$, which are connected into a long vector. The number of neurons in the input layer matches the size of the feature set. The number of neurons in the output layer determines the number of classifications, which in this case is two (statement of positive or negative). By using the gradient-based propagation, the weights of the edges are modified in the hidden layer at each point in time. After numerous testing and training epochs, the classification of sentiment is attained from the model.

F. Google Sentiment Analysis

The understanding of Natural language processing and the analysis of sentiments, analysis of entity, classifying of content, sentiment of entity, and analysis of the syntax are all offered by the Natural Language API from Google's Cloud [33]. The most recent NLP algorithm is called the bidirectional encoder representation (BERT), and it is a member of the Google's bigger family of cloud machine learning APIs.

G. Microsoft Azure Sentiment API

A text analytics API for improved NLP is available on the cloud platform Microsoft Azure. With four important features of analyzing the sentiments, extraction of key phrase, recognition of the named entity, and the detection of language, this open-source platform offers its services over raw text [34]. A range of AI and machine learning methods are accessible in the cloud for any project development because of Azure Cognitive Services, which offers the API as a member of its family.

H. IBM

The cutting-edge commercial artificial intelligence technology called IBM Watson is being currently made available [35]. Recent developments of global innovation for machine learning is supported by this open technology. As a member of its family, IBM Watson provides an API free for comprehending natural language and conducts sentiment analysis. Therefore this deep learning technology is built in cloud to investigate textual information that is complicated at various levels and classes.

I. Ensemble-Based Weighted Voting

The voting weights in weighted voting should vary for each classification between several outcome classes. The weight must be high for the specific class for which the categorization functions well. Finding the appropriate voting weights for each class in each classifier is very crucial. Multiple learning algorithms are used in voting-based ensemble approaches which strengthen the classification model. Weighted voting-based ensemble methods provide a more adaptable and fine-grained way to anticipate the actual output classes than the unweighted (majority) voting dependent ensemble methods. Bagging has low variance, hence it is not efficient with stable models; boosting leads to overfitting when it is not tuned properly. Therefore, voting ensemble produces a good balance among the variance and bias, this is efficient in various scenarios and can aid in decreasing overfitting. When related to single models, the ensemble approaches have a number of benefits such as increased performance and accuracy, especially for complicated and noisy issues. By employing various subsets and attributes of the data, as well as weighing the balance between variance and bias, the chances of overfitting and underfitting are also lessened. Multiple machine-learning algorithms are used in ensemble-based weighted voting for word embedding with sentiment analysis to categorize sentences into distinct sentiment categories of positive, neutral, or negative. Based on the way each algorithm categorizes the phrases, it is given a weight, and these weights are then used to determine an overall vote on the sentiment of a given word, phrase, or sentence. The individual predictions made by the algorithms weighted by their respective accuracies are combined to create the overall sentiment score. As a result of this, the sentiment analysis becomes more accurate because the consensus of several algorithms takes precedence over individual predictions. For each sample, the weighted voting prediction category is defined in Equation (7) as,

$$C_k = \underset{j}{\arg max} \sum_{i=1}^{D} \left(\Delta_{ij} \times w_i \right) \tag{7}$$

where C_k is category of weighted voting prediction, w_i is the weight of the i^{th} base classifier in an ensemble and Δ_{ij} is the binary variable. If the i^{th} base classifier classifies the sample k into the j^{th} category, then $\Delta_{ij} = 0$; otherwise, $\Delta_{ij} = 1$.

By aggregating the single classifier's classification results and choosing the group with the highest overall vote based on the weights assigned to the individual classifiers, the ensemble technique of weighted voting is utilized to enhance the performance of the classification model.

J. Final Sentiment Prediction

A probability distribution over the sentiment classes (negative, neutral, and positive) often makes up the final sentiment prediction in the Word Embedding Attention and Balanced Cross Entropy technique. The likelihood that a text falls into a given sentiment category is indicated by the probability score that is given to each sentiment class. The class with the greatest probability sentiment score is picked to arrive at the final sentiment prediction. The weighted predictions of all the models are added up to get the final sentiment prediction. The final sentiment prediction is positive when it has the probability of the highest score for instance, when the probability distribution is [0.1, 0.2, 0.7] respectively, for neutral, negative, and positive attitudes. The final prediction is the aggregated or combined prediction made by the ensemble shown in Equation (8),

$$Final\,Prediction = \Sigma\left(w_i * P_i \right) / \sum w_i \tag{8}$$

where the prediction made by each model is denoted as P_i, the summation or total across all models is denoted as Σ and the weight assigned to each model's prediction is denoted as w_i.

IV. Results

The proposed sentiment analysis implementation is developed using Python 3.7, Numpy 1.16.2, PyTouch 1.1.0, NetworkX 2.4, SciPy 1.3.1, TensorFlow 2.1.0, and Scikit-Learn 0.21.3. A computer with 7th generation Intel Core i5 CPU, 8 GB GeForce RTX 2070, and RAM of 32GB is used for the tests. Preprocessing is done using the Word-NetLemmatizer, wordtokenize, and NumPy packages with the Jupyter module of the Anaconda framework. For feature extraction, Word2Vec, Pandas, and PCA libraries are utilized. The dataset from Amazon is used in this simulation and is split into two parts; 20% of it is used to test the classifier, while the other 80% is used to train the actual LSTM classifier. The overall setup is filled by layer size as 64 and 128, optimizer is Adam, the batch size of 64, and epoch is 500.

Word embedding and pre-processing procedures are carried out using both the training and testing data. From the test data, the embedded word characteristics are then fed to the trained classifiers. The sentiment score of the input comments is examined effectively by the trained classifiers. The performance of the classifier is evaluated based on its classification (negative, positive, and neutral) of the input comments with respect to precision, accuracy, f-score, and recall. The performance measurements are described in the section that follows.

A. Performance Metrics

The following common statistical parameters are utilized to evaluate the implemented model. Accuracy, precision, F-score, and recall are calculated by using equations (9), (10), (11), and (12).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{9}$$

$$Precision = \frac{TP}{TP+FP} \tag{10}$$

$$F-score = \frac{Precision * Recall}{Precision + Recall} \tag{11}$$

$$Recall = \frac{TP}{TP+FN} \tag{12}$$

where, FP and FN correspondingly denote false positives and false negatives, while TP and TN correspondingly denote true positives and true negatives. The sentiment score measures the amount of sentimental words in a sentence; “positive” words like “pleasant” and “perfect”, and “negative” words like “unhappy” and “disappointing” are used to indicate emotions.

TABLE I
SENTIMENT SENTENCES DISTRIBUTION

Datasets	*Positive*	*Neutral*	*Negative*
Reddit	1103 (17%)	3090 (21%)	8277 (22%)
Twitter US airline	2363 (16%)	1430 (22%)	4001 (61%)
Amazon	15,830 (42%)	13,142 (35%)	9178 (63%)

Sentiment analysis is an automated approach that interprets the thoughts or emotions hidden inside a text. It is among the most fascinating subfields within NLP, an area of AI that focuses on how computers understand language. Sentiment analysis examines the subjective data in a statement such as reviews, opinions, sentiment (feelings), or the attitude towards a subject, person, or thing. Reddit’s dataset is imbalanced with the leading sentiment being positive. Another imbalanced dataset is the Twitter US Airline, wherein majority of the data reflects negative statements, thus making it uneven. By using the Tweepy module, this dataset is collected from Twitter. Also, the Amazon dataset that is used to train the model is most similar to this one in nature. By comparing all three datasets, the Amazon dataset has a greater amount of values as shown in Table I. It is seen that the Reddit dataset contains 17% of positive, 21% of neutral, and 22% of negative distributions; while the Twitter US airline dataset contains 16% positive, 22% neutral and 61% negatives. However, in the amazon dataset, 42% of positive, 35% of neutral and 63% negative are observed in its sentiment sentence distribution. The Amazon dataset’s nature which is similar to the training data and has a wider scope is responsible for this superior performance. When compared to LSTM, the ensemble method is more robust. The implemented weight ensemble with LSTM is utilized to enhance the performance of generalization, and to decrease the overfitting which achieves high performance analysis. Table II shows the performance analysis of the actual LSTM and weighted ensemble-based LSTM model, while Table III shows the comparative analysis of the proposed method in terms of various performance metrics.

TABLE II
PERFORMANCE ANALYSIS OF LSTM AND WEIGHTED ENSEMBLE BASED LSTM

Architecture	*Accuracy (%)*	*Precision (%)*	*F-score (%)*	*Recall (%)*
LSTM Without Weighted Ensemble	95.26	94.49	94.03	93.45
Weighted ensemble with LSTM	99.82	99.4	99.02	99.7

TABLE III
THE IMPLEMENTED SYSTEM COMPARISON WITH EXISTING SYSTEMS IN TERMS OF ACCURACY, PRECISION, F-SCORE, AND RECALL

Architecture	*Dataset*	*Accuracy (%)*	*Precision (%)*	*F-score (%)*	*Recall (%)*
APSO-LSTM [24]	Amazon review dataset	96.8	85.28	80.45	76.08
B-MLCNN [29]		95	94	94	93
Weighted ensemble with LSTM		99.82	99.4	99.02	99.7

The proposed model’s performance is tested based on the parameters of precision, accuracy, recall, and f-score. It is compared to the currently available approaches with regard to precision, accuracy, recall, and f-score as mentioned in the comparison study displayed in Table III. It is observed in Table III that the implemented model outperforms all other models, including LSTM+FL, APSO-LSTM, ABCDM, SSentiA, and Hybrid CNN+LSTM model. The current model of Weighted ensemble with LSTM obtains the highest accuracy. Figure 4 graphically shows the comparative performance analysis of the implemented Weighted ensemble with LSTM model alongside the existing methods namely, APSO-LSTM [24], and B-MLCNN [29]. The deliverance of the suggested technique is higher as it provides better results due to effective sentiment analysis using word embedding attention and balanced cross entropy technique.

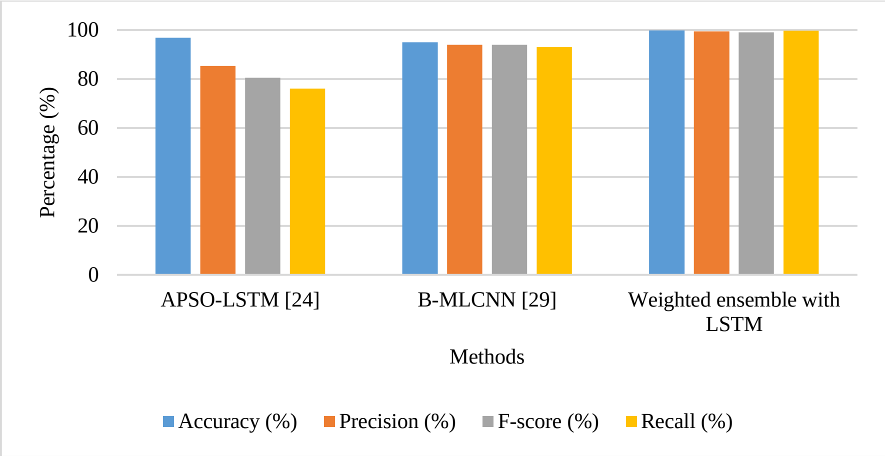

Fig. 4. Graphical representation of the implemented system comparison with existing systems

B. Discussion

The goal of this study is to develop a weighted ensemble with LSTM and a specialised deep learning model using unique word embedding approaches to enhance the sentiment analysis performance. The weighted ensemble with LSTM achieves superior outcomes because of its excellent generalization capabilities when contrasted with the existing methods such as APSO-LSTM [24] and B-MLCNN [29]. The proposed weighted ensemble with LSTM achieves 99.82 % accuracy, 99.4% precision, 99.02% f-score, and 99.7% recall in sentiment analysis, which are much greater than those of the existing methods. LSTMs are superior to conventional RNNs in a number of ways. To start with, they are far more adept at managing long-term dependence. This is because of their propensity for long-term memory retention. Secondly, the vanishing gradient problem significantly lessens the vulnerability of LSTMs. But in order to learn efficiently, they need more training data and are more complex than the conventional RNNs. Secondly, they are not appropriate for online learning tasks when the input data is not a sequence, as in the case of prediction or classification tasks. Third, training the LSTMs on big datasets is a time consuming process.

V. Conclusion

The large number of products available on e-commerce websites occasionally overwhelms the buyers and makes it challenging for them to locate the ideal item. Due to the increased competition amongst international commercial sites, it is more important than ever to operate profitably. By making it easier for users to identify the right products based on their interests, the recommendation systems seek to enhance the performance of e-commerce platforms. Numerous algorithms for recommendation systems are used for this purpose. Here, an efficient framework for sentiment analysis is implemented in which a model of ensemble deep learning language creates a sentiment analysis network that is weighted collectively using an LSTM. This model's classification performance is compared to the most recent methods employing sentiment analysis. This implemented model was tested on Amazon dataset to predict user attitudes by analyzing their tweets. The outcomes indicated that this ensemble approach for sentiment analysis proved to be useful. Because of its strong generalization capabilities, the weighted ensemble with LSTM produced significantly good results. The precision, accuracy, recall, and f-score metrics were all used in the performance evaluation of the implemented method. The respective results of these metrics were measured at the maximum for the proposed methodology, with the values being 99.4%, 99.82%, 99.7%, and 99.02%. To improve the classification performance, the model can be upgraded by adding a number of supplementary attributes in the future work.

References

[1] M. Singh, A. K. Jakhar, and S. Pandey, "Sentiment analysis on the impact of coronavirus on social life using the BERT model," *Social Network Anal. Min.*, vol. 11, no. 1, p. 33, Dec. 2021.

[2] M. Qorich and R. El Ouazzani, "Text sentiment classification of Amazon reviews using word embeddings and convolutional neural networks," *The Journal of Supercomputing*, pp. 1-26, Jul. 2023.

[3] A. Kumar, P. Gupta, R. Balan, L. B. M. Neti, and A. Malapati, "Bert based semi-supervised hybrid approach for aspect and sentiment classification," *Neural Process. Lett.*, vol. 53, no. 6, pp. 4207-4224, Dec. 2021.

[4] I. Priyadarshini and C. Cotton, "A novel LSTM–CNN–grid search-based deep neural network for sentiment analysis," *The Journal of Supercomputing*, vol. 77, no. 12, pp. 13911-13932, Dec. 2021.

[5] Y. Y. Tan, C. O. Chow, J. Kanesan, J. H. Chuah, and Y. Lim, "Sentiment Analysis and Sarcasm Detection using Deep Multi-Task Learning," *Wireless Pers. Commun.*, vol. 129, no. 3, pp. 2213-2237, Apr. 2023.

[6] J. Sangeetha, and U. Kumaran, "A hybrid optimization algorithm using BiLSTM structure for sentiment analysis," *Meas.: Sens.*, vol. 25, p. 100619, Feb. 2023.

[7] Y. Zhang, J. Wang, and X. Zhang, "Personalized sentiment classification of customer reviews via an interactive attributes attention model," *Knowledge-Based Syst.*, vol. 226, p. 107135, Aug. 2021.

[8] K. L. Tan, 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, Feb. 2022.

[9] J. Chen, J. Yu, S. Zhao, and Y. Zhang, "User's review habits enhanced hierarchical neural network for document-level sentiment classification," *Neural Process. Lett.*, vol. 53, no. 3, pp. 2095-2111, Jun. 2021.

[10] P. Rakshit, P. K. Srivastava, M. Afjal, and S. K. Srivastava, "Sentimental Analytics on Indian Big Billion Day of Flip Kart and Amazon," *SN Comput. Sci.*, vol. 2, no. 3, p. 204, May 2021.

[11] J. Liu, S. Zheng, G. Xu, and M. Lin, "Cross-domain sentiment aware word embeddings for review sentiment analysis," *Int. J. Mach. Learn. Cybern.*, vol. 12, no. 2, pp. 343-354, Feb. 2021.

[12] Y. Zhang, Z. Liu, and C. Sang, "Unifying paragraph embeddings and neural collaborative filtering for hybrid recommendation," *Appl. Soft Comput.*, vol. 106, p. 107345, Jul. 2021.

[13] F. Huang, X. Li, C. Yuan, S. Zhang, J. Zhang, and S. Qiao, "Attention-Emotion-Enhanced Convolutional LSTM for Sentiment Analysis," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 33, no. 9, pp. 4332-4345, Sep. 2022.
[14] Y. Cai, Q. Huang, Z. Lin, J. Xu, Z. Chen, and Q. Li, "Recurrent neural network with pooling operation and attention mechanism for sentiment analysis: A multi-task learning approach," *Knowledge-Based Syst.*, vol. 203, p. 105856, Sep. 2020.
[15] S. T. Kokab, S. Asghar, and S. Naz, "Transformer-based deep learning models for the sentiment analysis of social media data," *Array*, vol. 14, p. 100157, Jul. 2022.
[16] O. Gokalp, E. Tasci, and A. Ugur, "A novel wrapper feature selection algorithm based on iterated greedy metaheuristic for sentiment classification," *Expert Syst. Appl.*, vol. 146, p. 113176, May 2020.
[17] Q. Lu, Z. Zhu, D. Zhang, W. Wu, and Q. Guo, "Interactive rule attention network for aspect-level sentiment analysis," *IEEE Access*, vol. 8, pp. 52505-52516, Mar. 2020.
[18] M. Acharya, S. Yadav, and K. K. Mohbey, "How can we create a recommender system for tourism? A location centric spatial binning-based methodology using social networks," *Int. J. Inf. Manage. Data Insights*, vol. 3, no. 1, p. 100161, Apr. 2023.
[19] K. K. Mohbey, "Sentiment analysis for product rating using a deep learning approach," in *2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)*, IEEE, 2021, pp. 121-126.
[20] K. K. Mohbey, S. Kumar, and V. Koolwal, "Advertisement prediction in social media environment using big data framework," in *S. Tanwar, S. Tyagi, N. Kumar, (eds): Multimedia Big Data Computing for IoT Applications: Concepts, Paradigms and Solutions, Intelligent Systems Reference Library*, vol. 163, Springer, Singapore, 2020, pp. 323-341.
[21] G. Meena, K. K. Mohbey, and A. Indian, "Categorizing sentiment polarities in social networks data using convolutional neural network," *SSN Comput. Sci.*, Vol.3, No.2, p. 116, Mar. 2022.
[22] K. K. Mohbey, G. Meena, S. Kumar, and K. Lokesh, "A CNN-LSTM-Based Hybrid Deep Learning Approach for Sentiment Analysis on Monkeypox Tweets," *New Gener. Comput.*, pp. 1-9, Aug. 2023.
[23] M. Sivakumar and S. R. Uyyala, "Aspect-based sentiment analysis of mobile phone reviews using LSTM and fuzzy logic," *Int. J. Data. Sci. Anal.*, vol. 12, no. 4, pp. 355-367, Oct. 2021.
[24] J. Shobana and M. Murali, "An efficient sentiment analysis methodology based on long short-term memory networks," *Complex Intell. Syst.*, vol. 7, no. 5, pp. 2485-2501, Oct. 2021.
[25] M. E. Basiri, S. Nemati, M. Abdar, E. Cambria, and U. R. Acharya, "ABCDM: An attention-based bidirectional CNN-RNN deep model for sentiment analysis," *Future Gener. Comput. Syst.*, vol. 115, pp. 279-294, Feb. 2021.
[26] C. Zhao, S. Wang, D. Li, X. Liu, X. Yang, and J. Liu, "Cross-domain sentiment classification via parameter transferring and attention sharing mechanism," *Inf. Sci.*, vol. 578, pp. 281-296, Nov. 2021.
[27] S. Sazzed and S. Jayarathna, "Ssentia: a self-supervised sentiment analyzer for classification from unlabeled data," *Mach. Learn. Appl.*, vol. 4, p. 100026, Jun. 2021.
[28] A. Dadhich and B. Thankachan, "Social & juristic challenges of AI for opinion mining approaches on Amazon & flipkart product reviews using machine learning algorithms," *SN Comput. Sci.*, vol. 2, p. 180, Mar. 2021.
[29] P. Atandoh, F. Zhang, D. Adu-Gyamfi, P. H. Atandoh, and R. E. Nuhoho, "Integrated deep learning paradigm for document-based sentiment analysis," *Journal of King Saud University-Computer and Information Sciences*, vol. 35, no. 7, p. 101578, 2023.
[30] N. K. Gondhi, E. Sharma, A. H. Alharbi, R. Verma, and M. A. Shah, "Efficient long short-term memory-based sentiment analysis of e-commerce reviews," *Comput. Intell. Neurosci.*, vol. 2022, p. 3464524, Jun. 2022.
[31] Z. Al-Makhadmeh and A. Tolba, "Automatic hate speech detection using killer natural language processing optimizing ensemble deep learning approach," *Computing*, vol. 102, pp. 501-522, Feb. 2020.
[32] T. Swathi, N. Kasiviswanath, and A. A. Rao, "An optimal deep learning-based LSTM for stock price prediction using twitter sentiment analysis," *Appl. Intell.*, vol. 52, no. 12, pp. 13675-13688, Sep. 2022.
[33] 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, Feb. 2022.
[34] T. Ermakova, B. Fabian, E. Golimblevskaia, and M. Henke, "A Comparison of Commercial Sentiment Analysis Services," *SN Comput. Sci.*, vol. 4, no. 5, p. 477, Jun. 2023.
[35] M. Cherradi, F. Bouhafer, and A. E. Haddadi, "Data lake governance using IBM-Watson knowledge catalog," *Sci. Afr.*, vol. 21, p. e01854, Sep. 2023.

Prashant Sharma is working as an Associate Professor in Computer Science and Engineering Department, Faculty of Engineering, Mandsaur University, Mandsaur (MP).

Vijaya Sagvekar is currently pursuing PhD (Computer Engineering) at PAHER University, Udaipur (Rajasthan) and also working as an Assistant Professor in Information technology department at Vasantdada Patil college of engineering and visual arts, Mumbai. She has teaching, research experience of 12 years. Her area of interest are Database management systems, Software engineering, Web Mining, Data Optimization, Network security.