Bi-Directional CNN-RNN Architecture with Group-Wise Enhancement and Attention Mechanisms for Cryptocurrency Sentiment Analysis

Gül Cihan Habek, Mansur Alp Toçoğlu & Aytuğ Onan

Published online: 13 Nov 2022.

Submit your article to this journal

Article views: 979

View related articles

View Crossmark data

Citing articles: 1 View citing articles

Taylor & Francis
Taylor & Francis Group

Bi-Directional CNN-RNN Architecture with Group-Wise Enhancement and Attention Mechanisms for Cryptocurrency Sentiment Analysis

Gül Cihan Habek[a], Mansur Alp Toçoğlu [b], and Aytuğ Onan[b]

[a]Department of Computer Engineering, Karamanoğlu Mehmetbey University, Karaman, Turkey;
[b]Department of Computer Engineering, İzmir Katip Çelebi University, İzmir, Turkey

ABSTRACT

As the cryptocurrency trading market has grown significantly in recent years, the number of comments related to cryptocurrency has increased tremendously in social media platforms. Due to this, sentiment analysis of the cryptocurrency-related comments has become highly desirable to give a comprehensive picture of peoples' opinions about the trend of the market. In this regard, we perform cryptocurrency-related text sentiment classification using tweets based on positive and negative sentiments. For increasing the efficacy of the sentiment analysis, we introduce a novel deep neural network hybrid architecture which is composed of an embedding layer, a convolution layer, a group-wise enhancement mechanism, a bidirectional layer, an attention mechanism, and a fully connected layer. Local features are derived using a convolution layer, and weight values associated with intuitive features are developed using the group-wise enhancement mechanism. After feeding the improved context vector to the bidirectional layer to grab global features, the attention mechanism and the fully connected layer have been employed. The experimental findings indicate that the proposed architecture outperforms the state-of-the-art architectures with an accuracy value of 93.77%.

Introduction

The popularity of trading cryptocurrency coins in trading platforms has increased exponentially in recent years. Cryptocurrency is a blockchain-based digital asset which is used as peer to peer digital exchange units in online payments. This digital currency is basically based on two encryption algorithms, which are elliptic curve encryption and public-private key pairs. Besides, in order to secure online payments, hashing functions are used to provide legitimate and unique transactions (Aslam et al. 2022). Bitcoin is the first cryptocurrency coin which is introduced in 2009. It is designed to be an independent digital asset against the currency tracking systems of

CONTACT Mansur Alp Toçoğlu ✉ mansuralp.tocoglu@ikc.edu.tr 🖂 Department of Computer Engineering, İzmir Katip Çelebi University, İzmir, Turkey

governments and banks by hiding the information of the sender and the receiver in a payment process (Di Pierro 2017). There are well-known cryptocurrencies in the market. Ethereum is one of them which was introduced in 2015 by Vitalik Buterin (Tikhomirov et al. 2018). Ethereum enables its users to create smart contracts of their own (Wöhrer and Zdun 2018).

In today's world, there are no more physical boundaries for sharing information. As the usage rates of communication technologies evolve very rapidly, people communicate with each other using social media platforms and microblogging sites such as Twitter, Facebook, Instagram. Here, such platforms play an important role for people to share their comments, feedback, and observations about any kind of topic with a much larger audience. As a result of the popularity of the cryptocurrencies, increasing number of people have started to write articles on trading point view, where the feelings and the thoughts are instantly shared on online platforms with short sentences consisting of a few words rather than long texts. Therefore, sentiment analysis of the comments on cryptocurrency emerges as an important field of study.

The cryptocurrency market is considered to be volatile where the prices of the cryptocurrencies move up and down frequently. These fluctuations in the market provide investors the opportunity to earn money. For this purpose, investors use many tools to forecast the direction of the market. On the other hand, governmental policies and general public opinions have efficacy on the market. In this regard, sentiment analysis can be an effective approach to determine rise and fall of the market since it is considered as a valuable feedback information for cryptocurrency investment these days (Abraham et al. 2018; Colianni, Rosales, and Signorotti 2015). According to the studies (Lamon, Nielsen, and Redondo 2017; Wołk 2020), tweets containing sentiments play an important role for influencing investor demands. As a result, analyzing peoples' sentiments to predict the directions of the cryptocurrency markets has become an important classification task (Chuen, Guo, and Wang 2017).

The aim of this study is to contribute Turkish text sentiment analysis studies related to cryptocurrency in the literature by providing a novel supervised learning model which increases the performance of identification of sentiments related to cryptocurrency coins in social media platforms. To do this, a new Turkish cryptocurrency-related dataset is created which is composed of 9,548 tweets in total where 5,907 tweets are labeled as positive and the rest 3,641 tweets are labeled as negative. After that, a new hybrid deep neural network architecture is presented which is based on Convolutional Neural Network (CNN), group-wise enhancement mechanism, Recurrent Neural Network (RNN) and attention mechanism. Local features have been derived using a convolution layer in this scheme. Then, using the group-wise enhancement mechanism, an enhanced context vector was obtained by assigning a higher weight to informative features and a lower weight to uninformative

features. The enhanced context vector was then fed into bidirectional Long Short-term Memory (LSTM) architecture to capture global features. Then, the attention mechanism has been employed so that the different weight values can be assigned to global features to reflect the importance regarding the contextual information. Finally, the fully connected layer has been utilized to identify sentiment orientation of text documents. We examined the predictive performance of three different versions of the proposed model. In addition, the prediction performances of the three word embedding schemes were evaluated together with the four basic deep neural networks and the proposed model. The comprehensive empirical analysis on two sentiment classification datasets with the state-of-the-art architectures indicates that the proposed scheme can yield promising results on sentiment analysis tasks. The contributions of the paper can be summarized as follows:

- Motivated by the encouraging findings obtained by the hybrid architectures based on CNNs and RNNs on NLP tasks, we introduce a bidirectional CNN-RNN architecture for text sentiment classification, along with the group-wise enhancement and the attention mechanisms.
- To the best of our knowledge, this study is the first comprehensive analysis of the state-of-art deep neural architectures for sentiment analysis on cryptocurrency.
- In addition to the analysis on fourteen architectures, we also present a hybrid deep neural architecture with high predictive performance. To the best of our knowledge, this is the first hybrid architecture in which group-wise enhancement mechanism in conjunction with attention mechanism.

The remaining sections are arranged as follows: The existing works are presented in the Related Works Section. The Theoretical Foundations Section includes a brief overview of the proposed model's fundamental components. The next section addresses the proposed model for sentiment analysis. The empirical findings are discussed in the Experiments and Results Section. In the last part, the study is concluded.

Related Works

As the usage rates of social media platforms have grown very rapidly, identifications of sentiments and emotions in text data have become an important research area in the literature (Calefato, Lanubile, and Novielli 2017; Haryadi and Kusuma 2019; Hasan, Rundensteiner, and Agu 2019; Onan 2021, 2022; Salam and Gupta 2018; Shah et al. 2019; Tocoglu, Ozturkmenoglu, and Alpkocak 2019; Zhang et al. 2020). However, identification of sentiments about cryptocurrency trading has not evolved at the same pace since it is

a new concept. In the literature, the studies related to cryptocurrency market are generally focused on price prediction problems by examining correlations between price and sentiment values of cryptocurrency coins using machine learning techniques.

In the study, the authors (Mehta et al. 2020) used sentiment analysis techniques to make price prediction for Bitcoin cryptocurrency price in real-time using Twitter and news data. To accomplish this goal, they trained their LSTM model with a dataset composed of 1559 tuples where each tuple has date, sentiment values and Bitcoin price data. The suggested model is live on a server where it makes predictions of Bitcoin price continuously. In another study (Raju and Tarif 2020), the authors focused on providing informative predictions about future Bitcoin market price by examining the correlation between Bitcoin price and the sentiments of tweets about it. They used LSTM architecture to construct their prediction model.

Şaşmaz and Tek (2021) focused on sentiment analysis of NEO cryptocurrency coin and the correlation of daily sentiment of tweets with the market price of NEO. For this, they first gathered a dataset composed of last five years of daily tweets regarding NEO. Next, the raw dataset was labeled manually for three sentiment categories which are positive, negative and neutral. Sentiment analysis experiments were performed by constructing models using Random Forest and BERT classifiers where the former algorithm outperformed the other. In the study's second phase, the daily sentiment of tweets was compared to the market price of NEO.

Valencia, Gómez-Espinosa, and Valdés-Aguirre (2019) sought to anticipate the price fluctuations of the cryptocurrencies Bitcoin, Litecoin, Ethereum, and Ripple. To begin, they gathered a dataset of tweets and market price caps for four cryptocurrencies. Then, they compared the predictive performances of several classifiers. In general, the model constructed using multi-layer perceptron algorithm performed the highest result among others.

Pant et al. (2018) concentrated on forecasting the Bitcoin price in the market using sentiment analysis methods. Firstly, they gathered tweets from different news accounts and categorized them manually as positive and negative. Next, they trained a voting classifier model, consists of five algorithms, using the categorized dataset to make daily sentiment percentage predictions. The accuracy value for the voting classifier was achieved as 81.39%. Further, the authors combined the output of the voting classifier with the historical price of Bitcoin and used them to feed the RNN model which predicts Bitcoin price with an accuracy rate of 77.62%.

Köksal et al. (2021) conducted a sentiment analysis using Naïve Bayes and Logistic Regression classifiers with 3,737 tweets related to Bitcoin in the first phase of their study. They labeled the tweets in the dataset for three sentiments which are positive, negative and neutral. As a result, Logistic Regression classifier performed the highest accuracy value. In the second phase of the study, the

authors estimated the market closing price of Bitcoin by using opening price of Bitcoin and the rate of daily positive tweets containing the keyword Bitcoin. To do so, they used Linear and Random Forest Regression algorithms.

Çılgın et al. (2020) evaluated the correlation between tweets and Bitcoin price. They first gathered 2,819,784 tweets in English containing the keyword Bitcoin. Then, 1,500 tweets were chosen and labeled as positive, negative and neutral to be used as the training dataset. The authors used several conventional classifiers to construct the sentiment analysis model. In the next step, the rest of the raw dataset was classified by using the trained model. Then, the authors conducted a correlation analysis using the rates of the positive tweets sent for each day and the Bitcoin daily closing prices.

Huang et al. (2021) focused on analyzing the correlation between price movements in the market and sentiments in Chinese social media platform Sina-Weibo for cryptocurrency price prediction. The authors created crypto-based sentiment dictionary and proposed a machine learning approach using LSTM architecture for the prediction of price trend of the cryptocurrency market. The proposed method outperformed the state of the art models in precision and recall by 18.5% and 15.4%, respectively.

The primary goal of the studies discussed above is to predict cryptocurrency market prices rather than conducting sentiment analysis itself. As a result, studies focusing on increasing the performances of sentiment analysis models for cryptocurrencies are limited. For this reason, this study proposes a novel hybrid deep neural network architecture for sentiment analysis on cryptocurrency-related tweets. In the literature, there are few studies which are mainly concentrated on sentiment analysis for cryptocurrency market. For example, Aslam et al. (2022) proposed a stacked deep learning ensemble architecture using LSTM followed by GRU. They used TextBlob and Text2Emotion libraries to categorize the raw tweet dataset. Considering the overall results in the study, the proposed architecture outperformed the compared models. On the other hand, in this study, we focused on enhancing the classification performance of the proposed model by using group-wise enhancement and attention mechanisms to reveal the informative features more precisely.

Theoretical Foundations

The basic elements are briefly described in this section. We summarized Word2vec word embedding scheme, and covered the deep neural network architectures in the following subsections.

Word2vec Model

The Vector Space Model (VSM) is not capable of capturing the semantic relationships between features of the dataset. VSM suffers in performance

due to problems of high dimensionality and sparsity. To handle such problems, neural language models have become popular in text mining tasks. In these models, a real-valued vector is generated for each word by encoding words with similar meanings in the vector space. Such representation is called dense vector representation which provides low dimensionality and text documents with semantic properties.

The Word2vec is a neural language model which builds word embeddings using a layered structure in sequence (Mikolov et al. 2013; Onan and Toçoğlu 2021). By requiring less manual preprocessing of texts with semantic and syntactic features, neural language models provide robust representation. The dense vector representation yields promising results for natural language tasks. The word2vec model is composed of three layers: an input layer, an output layer, and a hidden layer. It is an artificial neural network-based scheme for word embedding (Mikolov et al. 2013). It attempts to learn to embed words by calculating the probability that a given word is rooted in other words. The model is composed of two fundamental architectures: the skip-gram (SG) and the continuous bag of words (CBOW). The CBOW architecture defines the target word by using the content of each word as an input; on the other hand, the SG architecture anticipates the words surrounding the target word by using the objective words as an input. The CBOW architecture can function well with a small amount of data (Rasool et al. 2021). The SG architecture performs admirably on large datasets. Let we denote a sequence of training words $w_1, w_2, \ldots, w_T$ with length T, the objective of skip-gram model is determined based on Equation (1) (Mikolov et al. 2013):

$$arg\, max_\theta \frac{1}{T} \sum_{t=1}^{T} \sum_{-C \leq j \leq C, j \neq 0} log P_\theta\left(w_{t+j} | w_t\right) \qquad (1)$$

where C represent the size of training context, $P\left(w_{t+j} | w_t\right)$ represents a neural network with a set of parameters denoted by θ.

Deep Neural Network Architectures

In this section, we shared brief overviews of the four well-known deep learning architectures.

Convolutional Neural Networks

The convolutional neural network (CNN) is a specialized deep learning architecture that uses a mathematical operation named convolution to process input data. In text mining techniques, CNN is a well-known multi-layer network employed for identifying local features (Gutiérrez et al. 2018; Li and Wu 2015).

Recurrent Neural Networks

A recurrent neural network (RNN) is a deep learning architecture which trains on sequential and time-series data (Hochreiter and Schmidhuber 1997). In contrast to conventional feedforward neural networks, RNN architecture processes sequences using feedback loops to retain memory over time. Recurrent units in the architecture have generic configurations with no memory units or extra gates. The input length of the RNN architecture was used to calculate the time step size.

Long Short-Term Memory

The LSTM architecture is based on RNN architecture which has feedback connections, unlike conventional RNN (Rojas-barahona 2016). To cope with the vanishing gradient problem of RNN architecture, the LSTM algorithm employs a recurrent unit that contains a cell and input gate, output gate, and forget gate in total. These gates provide LSTM architectures to have the ability to retain long-term memory. Based on the cell structure, the information to be kept has been identified (Chung et al. 2014).

Gated Recurrent Unit

The gated recurrent unit (GRU) is a class of RNN architecture that also utilizes a gating mechanism similar to the LSTM model (Cho et al. 2014). However, because it only has two gates and requires fewer training parameters than LSTM design, GRU is less complicated.

Proposed Bi-Directional CNN-RNN Architecture (CGWELSTM)

To classify text sentiment, we introduce a new deep neural network architecture (CGWELSTM) in this study. Convolutional layers and large short-term memory are used in the CGWELSTM architecture to extract local and global properties from text sources. Figure 1 shows the outline of the CGWELSTM. The suggested technique is made up of seven main modules: an input layer, an embedding layer, a convolution layer, a group-wise enhancement mechanism, a bidirectional layer, an attention mechanism, and a fully connected layer.

The dense vector representation of text documents was first created by the embedding layer using deep neural language models. To extract useful local characteristics, the proposed technique uses a convolutional layer. The local features acquired by the convolutional layer were subjected to the group-wise enhancement technique to improve the relevant features. In this way, important features have been given greater importance coefficients while decreasing the coefficients associated with the other features. Local features have been split up into several categories in this approach. Bidirectional LSTM has been used to analyze the augmented context feature vector further to extract comprehensive features.
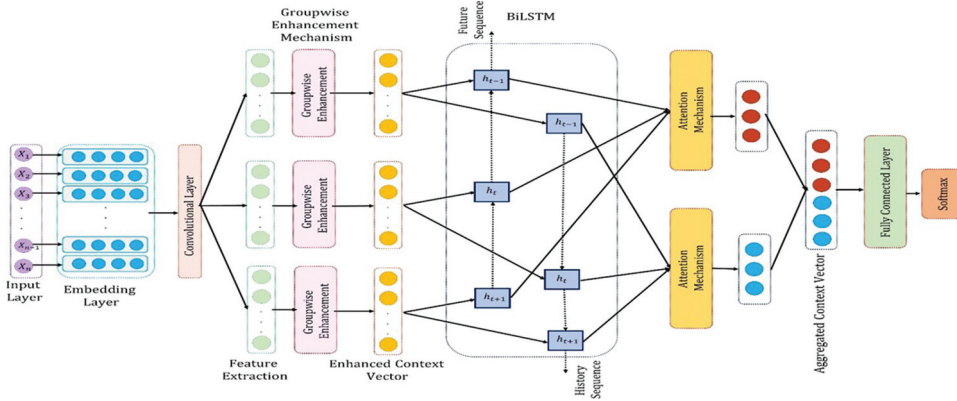
Figure 1. The proposed model.

Then, the attention method has been applied to the features, allowing for the assignment of various weights that affect a text's sentiment. The sentiment orientation of text documents was lastly determined utilizing a fully linked layer and an aggregated context feature set. Details on the CGWELSTM architecture's parts are presented in the remaining paragraphs of this section.

The Embedding Layer

The embedding layer module generates word embeddings from text using a pre-trained word2vec embedding matrix. We begin by transforming each word in an n-dimensional text document into its corresponding V-dimensional word vector and obtaining the word embedding matrix. In this way, the word embedding matrix $X = [x_1, x_2, \ldots, x_n] \in R^{n \times V}$ has been obtained.

The Convolutional Layer

We apply convolutions on the word embedding matrix in the convolution layer module to extract local features. CNN obtains the local context features on the text documents at this phase. Let $F^j \in R^{w \times V}$ corresponds to the filter, with w denoting the size of the sliding window. Concatenating the initial dimensions of the associated vectors yields the following equation, which has been used to derive the local context characteristics for w words:

$$m^j = \left[m_1^j, \ldots, m_i^j, \ldots, m_{L-w+1}^j \right] \qquad (2)$$

$$m_i^j = f\left(x_{i:i+w-1} \otimes F^j + b_0 \right] \qquad (3)$$

where f corresponds to the RELU and $\otimes$ corresponds to the convolution. Using a succession of identical-sized filters, local CNN architectural features were retrieved from the text file. On enhanced context, the convolution operation was repeated to produce the local feature matrix.

The Group-Wise Enhancement Mechanism

Utilizing feature weighting algorithms on the local features obtained by the convolutional layer in the group-wise enhancement mechanism, informative features were recovered. The spatial group-wise enhancement technique, which was first created for picture classification problems, is a variation of the module that we're talking about today (Li, Hu, and Yang 2019; Onan 2022). The mechanism is intended to make weight values for informative features stronger and weaker relative to uninformative feature weight values. Finding suitably distributed feature sets for developing deep neural architecture-based learning models is typically challenging due to the amount of uninformative information. Utilizing all the semantic data in important areas is the goal of the group-wise enhancement process. In this way, semantic vector has been obtained.

The word embedding matrix $X = [x_1, x_2, \ldots, x_n] \in R^{n \times V}$ has V dimensions. Here, n denotes the size of the text. Initially, we divide the representation scheme into M groups, obtaining $M^k = [x_1^k, x_2^k, \ldots, x_n^k]$, where $x_i \in R^{n/M}$, n=V, and k corresponds to the number of groups. The group's semantic vector has been obtained by using the following equation:

$$g^k = \frac{1}{n} \sum_{i=1}^{n} x_i^k \tag{4}$$

We can then determine the significance of each feature. The following equation determines the significance of features based on the similarity with the global semantic feature g^k and the local feature x_i^k using a dot product:

$$c_i^k = g^k . x_i^k \tag{5}$$

We normalize c over space c_i^k using the following equations to prevent biased values for samples:

$$c_i^{*k} = \frac{c_i^k - \mu_c^k}{\sigma_c^k + \varepsilon} \tag{6}$$

$$\mu_c^k = \frac{1}{n} \sum_j^n c_j^k \tag{7}$$

$$\left(\sigma_c^k\right)^2 = \frac{1}{n} \sum_j^n (c_j^k - \mu_c^k)^2 \tag{8}$$

where ε is a constant (assigned to 1e-5) that has been added for regularization. Moreover, γ, β have been added for each coefficient $c_i^* k$:

$$a_i{}^k = \gamma c_i{}^{*k} + \beta \tag{9}$$

where the parameter values γ, β have been determined empirically (Liu and Guo 2019). At the end, the improved vector $x_i^* k$ was constructed by using the following equation:

$$x_i{}^{*k} = x_i^k . \sigma\left(a_i{}^k\right) \tag{10}$$

The final features $M^{*k} = \left[x_1{}^{*k}, x_2{}^{*k}, \ldots, x_n{}^{*k}\right]$, where $x_i{}^k \in R^{n/M}$ have been obtained.

The Bidirectional Layer

In this layer, we have employed a bidirectional LSTM. Bidirectional RNN-based designs can also help in the capture of top-notch features by processing data in two directions. The following equations demonstrate that the context therefore includes both the past and the future:

$$c_t{}^{\rightarrow}, h_{t_{LSTM}}{}^{\rightarrow} = LSTM^{\rightarrow}\left(c_t{}^{\rightarrow}, h_{t-1_{LSTM}}{}^{\rightarrow}, x_t{}^{\rightarrow}\right) \tag{11}$$

$$c_t{}^{\leftarrow}, h_{t_{LSTM}}{}^{\leftarrow} = LSTM^{\leftarrow}\left(c_t{}^{\leftarrow}, h_{t-1_{LSTM}}{}^{\leftarrow}, x_t{}^{\leftarrow}\right) \tag{12}$$

$$h_{t_{LSTM}} = [h_{t_{LSTM}}{}^{\rightarrow}, h_{t_{LSTM}}{}^{\leftarrow}] \tag{13}$$

where $h_{t_{LSTM}}$ denotes the hidden state vector and c_t denotes the cell state vector.

The Attention Mechanism

A distribution over the input representations can be produced by combining the attention mechanism with deep neural network topologies in a few natural language processing applications. Using the attention mechanism, one can assign different weights to features. The equations below demonstrate how attention mechanisms can be employed to assign weights to features:

$$\alpha_t = \frac{exp\left(v^T . u_t\right)}{\sum_t exp(v . u_t)} \tag{14}$$

$$S_{A_w} = \sum_t \alpha_t h_t \tag{15}$$

where u_t and h_t stand for vectors of the LSTM units and v is a trainable parameter.

The Fully Connected Layer

The first two layers of the fully connected layer employs RELU function and the last layer employs the softmax activation function. In this layer, L2-norm regularization has been employed. Here, the regularization coefficient is equal to 0.1 and the dropout parameter is equal to 0.30. The binary cross entropy function has been employed as the loss function.

Experiments and Results

This section contains information about the dataset used in the empirical analysis, the experimental procedure, and the experimental results obtained using deep neural network architectures.

Dataset

In this paper, we collected a raw dataset consists of 25,000 Turkish cryptocurrency-related tweets by using a social networking service scraper named SNScrape library. The tweets are gathered from April 20, 2021 to March 20, 2022. To do so, we fetched the tweets which are tagged to fourteen specific hashtags which are #avax, #avalanche, #bitci, #btc, #bitcoin, #chz, #chiliz, #eth, #ethereum, #solana, #xrp, #crypto, #nft, #defi. In the annotation process of the raw data, each tweet is labeled as positive, negative, neutral and not crypto-related by two annotators. After the annotation process, the size of the dataset declined to 9,548 tweets in total because we eliminated the tweets categorized as neutral and not crypto-related and the tweets which are no consensus on the labeling results of the annotators. As a result, 5,907 tweets are labeled as positive and the rest 3,641 tweets are labeled as negative. Next, we preprocessed the dataset for empirical research. Firstly, we removed tags, usernames and links, which are ineffective and meaningless data for training machine learning models, from each tweet. After that, we converted all letters to lowercase and eliminated numeric characters, extra spaces, and punctuation marks. In the next step, we performed Porter stemmer to normalize the dataset (Pedregosa et al. 2011). Lastly, a pre-defined NLTK Turkish stopword list was used to extract stopwords in the dataset. In addition, we assessed the predictive performance of the proposed architecture on a well-known sentiment classification benchmark (i.e., Sentiment140 dataset) in this study. It contains 1,600,000 tweets that have been automatically classified as positive or negative (Go, Bhayani, and Huang 2009).

Table 1. The hyperparameters for the proposed scheme.

Hyperparameter	Value
Number of epochs	[1–50]
Batch size	50
Embedding padding size	[40–200]
Number of CNN layers	4
Window size I of Convolutional kernel	[2,3]
Convolutional kernel bias	64
Number of Pooling	4
Pooling size	2
BiGRU layer output size	[49,256]
BiLSTM layer output size	[49,256]
Number of Dropout	3
Dropout rate	0.30
Number of batch normalization	2
Number of attention mechanisms	2
Attention with context	$w = (256 \times 256)$, $b = 256$, $u = 245$
Number of dense layer	3
Learning rate	0.0001
Optimizer	Gaussian approach
Activation function type	ReLu

Experimental Procedure

The deep neural network architectures utilized in the empirical investigation have been implemented and trained using Tensorflow and the Keras framework. To improve the predictive effectiveness of each model, we employ hyperparameter optimization. Hyperparameters were optimized using the Gaussian approach, which is based on Bayesian optimization, to achieve this. To ensure exact comparison, the same parameters have been utilized for ABCDM (Basiri et al. 2021). The RELU activation function was utilized for the first two layers of the fully connected layer, and the soft-max activation function was employed for the third and final layer of the fully connected layer. The convolution filters' sizes were set at 4 and 5. The CNN's hidden layer has a 100×100 dimension. For the mini batch, the sample size is 50, and the dropout coefficient is 0.30. To represent text documents, we looked into the word2vec approach with 300 dimensions. In Table 1, the hyper parameters for the proposed scheme has been summarized.

Baseline Models

In this study, we assessed the predictive performance of fourteen cutting-edge deep neural network designs against the architecture. The results were encouraging. The material below condenses the specifics of the underlying models utilized in the empirical analysis:

- ABCDM architecture: A bidirectional layer follows an embedding layer in the model's first stage. Parallel applications of bidirectional LSTMs and bidirectional GRUs were utilized to extract global features from text

documents in the bidirectional layer. Convolution and pooling layers are utilized next, and then the attention mechanism (Basiri et al. 2021).

- AC-BiLSTM architecture: A hybrid deep neural network with an attention mechanism is placed after a one-dimensional convolutional layer made up of varied filter sizes. Finally, the sentiment of review documents was ascertained using the fully connected layer (Liu and Guo 2019).
- AGCNN architecture: The attention-gated layer followed the convolutional layer in the model's progression. Maximum pooling was then employed, and then a fully connected layer (Liu et al. 2019) came after that.
- ARC architecture: The model is made up of five modules: a bidirectional GRU, an attention mechanism, a convolutional layer, a pooling layer, and a fully connected layer (Wen and Li 2018).
- ATTPooling architecture: An RNN with an attention mechanism based on CNNs makes up the model. The first thing CNN does is describe what makes the area unique. The attention mechanism was then used to weigh the features. Finally, a fully linked layer has been used, followed by an RNN with CNN-based attention (Usama et al. 2020).
- CAT-BiGRU architecture: Bidirectional GRUs are used in the model to include convolution and attention. Two attention layers and a convolutional input-embedding bidirectional GRU make up this system (Kamal and Abulaish 2022).
- CNN-GRU architecture: The model consists of a GRU recurrent unit, a max-pooling layer, and an embedding layer that made use of pre-trained vectors from word2vec (Wang et al. 2016).
- CRNN architecture: The input word vectors are subjected to regional CNN in this model, which treats each phrase as a region. Then, local attributes are diminished via max pooling. Finally, a linear decoder is utilized to determine future valence and excitement while an LSTM layer is employed to identify longer-term dependencies (Yang et al. 2016).
- HAN architecture: A sequence encoder, a word-based attention mechanism, a sentence-based attention mechanism, and a sentence-based attention mechanism make up the model's four modules. This technique uses bidirectional GRUs in the sequence encoder and phrase encoder modules (Yang et al. 2016).
- Improved word vectors (IWV) architecture: Pre-trained sentiment analysis word embedding techniques are more accurate with the help of this model. The model consists of a fully connected layer that is followed by three convolutional layers with maximum pooling (Rezaeinia et al. 2019).
- SS-BED architecture: This model seeks to extract semantic information from text sources using two simultaneous LSTM layers. To categorize the sentiment of review materials, a fully connected neural network with a single hidden layer was deployed (Chatterjee et al. 2019).

- Tree bi-LSTM architecture: For text classification, the model used a tree-based bidirectional LSTM architecture (Li et al. 2015).
- Tree LSTM architecture: The model uses a recursive tree-based architecture in place of the composition layers included in traditional LSTM blocks to understand text documents (Zhu, Sobhani, and Guo 2015).
- The weighted embedding layer, dropout layer, N-gram convolution layer (with 1-gram, 2-gram, and 3-gram convolutions), max-pooling layer, LSTM layer, and fully connected layer make up the WCNNLSTM architecture (Onan 2021).

Model Variations

In addition to the proposed scheme and the fourteen state-of-the-art architectures mentioned in advance, we have also considered three model variants. The following summarizes the details of the model variations used in the empirical analysis:

- CGWELSTM-v1 architecture: All the elements of the proposed schemes have been used in this model, except for the group-wise enhancement mechanism. In this manner, the bidirectional layer has been provided directly with the local information that the convolutional layer extracted. The remainder of the plan is identical to the proposed scheme.
- CGWELSTM-v2 architecture: All the elements of the proposed scheme have been used in this model, except for the attention mechanism. In this manner, the bidirectional layer's global feature extractions have been combined, and the fully connected layer has continued processing the combined context vector (without the attention method).
- CGWELSTM-v3 architecture: All of the elements of the suggested scheme have been used in this model, with the exception of the bidirectional layer. We have used bi-GRU in the bidirectional layer rather than bi-LSTM. The recommended methodology has been applied to the remaining modules in the same manner.

Experimental Results

In the empirical analysis, the prediction performance, classification accuracy, precision, recall, of the three-word embedding schemes were evaluated in relation to the five deep neural network architectures (i.e. CNN, GRU, LSTM, RNN, and CGWELSTM) and F-measure. In Tables 2–5, accuracies, precision, recall, and F-measure values received with the aid of using comparative schemes are presented, respectively. As may be visible from the

empirical outcomes indexed in Table 2, the proposed scheme (ie, CGWELSTM) outperforms conventional deep neural architectures based on accuracy. Among the deep neural architectures compared, CNN offers the lowest prediction overall performance even as GRU offers the best prediction overall performance. Regarding the overall performance of word embedding methods, GloVe offers the lowest prediction overall performance and the word2vec (skip-gram) version offers the best prediction overall performance. For the 2 vector sizes taken into consideration with inside the evaluation and the 3 dimensions of the projection layer, the vector length three hundred offers better prediction overall performance, and the excellent overall performance is carried out while the projection layer length is set to three hundred. For other metrics used for predictive overall performance, particularly CGWELSTM, outperform conventional deep neural networks in terms of precision, recall, and F-measure.

Figures 2–5 display the summary figures for deep neural architectures. The CGWELSTM architecture achieved the highest average predictive performance.

The proposed scheme (specifically, CGWELSTM) is compared with fourteen basic deep neural network architectures and three model variants for classification accuracy, precision, recall, and F-measure values. Table 5 summarizes the classification accuracy values obtained using the compared architectures. The results show that the proposed scheme (i.e, the CGWELSTM architecture) outperforms the most advanced architectures. The CGWELSTM

Table 2. Classification accuracy values of the conventional architectures and the proposed scheme.

Word Embedding	Vector size	Dimension of projection layer	CNN	RNN	LSTM	GRU	CGWELSTM
word2vec (Skip-gram)	200	100	84,9062	86,1300	86,5193	87,2704	92,3317
word2vec (Skip-gram)	200	200	85,3091	86,1505	86,9681	87,2791	92,5790
word2vec (Skip-gram)	300	100	85,5395	86,8367	87,0924	87,7042	92,6569
word2vec (Skip-gram)	300	300	85,8715	87,3834	88,5862	87,9131	93,7737
word2vec (CBOW)	200	100	84,1355	84,6919	86,3301	86,5933	91,3796
word2vec (CBOW)	200	200	84,3848	84,8301	86,3321	86,7954	91,5729
word2vec (CBOW)	300	100	84,4691	85,2159	86,4670	87,2066	91,8501
word2vec (CBOW)	300	300	84,6564	85,8096	86,4886	87,2103	92,0511
fastText (Skip-gram)	200	100	83,0755	83,9260	85,9995	86,0562	90,8553
fastText (Skip-gram)	200	200	83,4248	84,1585	86,1311	86,2161	90,9434
fastText (Skip-gram)	300	100	83,9981	84,1790	86,2756	86,2965	91,0908
fastText (Skip-gram)	300	300	84,1152	84,4471	86,2942	86,4113	91,2866
fastText (CBOW)	200	100	82,7116	83,5779	84,1253	85,0902	90,2029
fastText (CBOW)	200	200	82,7349	83,6589	85,1541	85,3837	90,5247
fastText (CBOW)	300	100	82,8614	83,8124	85,2146	85,4376	90,7180
fastText (CBOW)	300	300	82,9599	83,8509	85,6648	85,5584	90,7982
GloVe	200	100	81,4875	82,6678	82,8163	83,2925	88,0650
GloVe	200	200	81,5490	82,7000	83,0422	84,1660	88,9275
GloVe	300	100	82,2485	82,9684	83,5552	84,5301	89,5525
GloVe	300	300	82,4531	83,1941	83,9329	84,7005	90,0796

Table 3. Precision values of the conventional architectures and the proposed scheme.

| Word Embedding	Vector size	Dimension of projection layer	CNN	RNN	LSTM	GRU	CGWELSTM
word2vec (Skip-gram)	200	100	0,8576	0,8700	0,8739	0,8815	0,9326
word2vec (Skip-gram)	200	200	0,8617	0,8702	0,8785	0,8816	0,9351
word2vec (Skip-gram)	300	100	0,8640	0,8771	0,8797	0,8859	0,9359
word2vec (Skip-gram)	300	300	0,8674	0,8827	0,8948	0,8880	0,9472
word2vec (CBOW)	200	100	0,8499	0,8555	0,8720	0,8747	0,9230
word2vec (CBOW)	200	200	0,8524	0,8569	0,8720	0,8767	0,9250
word2vec (CBOW)	300	100	0,8532	0,8608	0,8734	0,8809	0,9278
word2vec (CBOW)	300	300	0,8551	0,8668	0,8736	0,8809	0,9298
fastText (Skip-gram)	200	100	0,8391	0,8477	0,8687	0,8693	0,9177
fastText (Skip-gram)	200	200	0,8427	0,8501	0,8700	0,8709	0,9186
fastText (Skip-gram)	300	100	0,8485	0,8503	0,8715	0,8717	0,9201
fastText (Skip-gram)	300	300	0,8496	0,8530	0,8717	0,8728	0,9221
fastText (CBOW)	200	100	0,8355	0,8442	0,8498	0,8595	0,9111
fastText (CBOW)	200	200	0,8357	0,8450	0,8601	0,8625	0,9144
fastText (CBOW)	300	100	0,8370	0,8466	0,8608	0,8630	0,9163
fastText (CBOW)	300	300	0,8380	0,8470	0,8653	0,8642	0,9172
GloVe	200	100	0,8231	0,8350	0,8365	0,8413	0,8895
GloVe	200	200	0,8237	0,8354	0,8388	0,8502	0,8983
GloVe	300	100	0,8308	0,8381	0,8440	0,8538	0,9046
GloVe	300	300	0,8329	0,8403	0,8478	0,8556	0,9099

Table 4. Recall values of the conventional architectures and the proposed scheme.

Word Embedding	Vector size	Dimension of projection layer	CNN	RNN	LSTM	GRU	CGWELSTM
word2vec (Skip-gram)	200	100	0,8664	0,8789	0,8828	0,8905	0,9422
word2vec (Skip-gram)	200	200	0,8705	0,8791	0,8874	0,8906	0,9447
word2vec (Skip-gram)	300	100	0,8729	0,8861	0,8887	0,8949	0,9455
word2vec (Skip-gram)	300	300	0,8762	0,8917	0,9039	0,8971	0,9569
word2vec (CBOW)	200	100	0,8585	0,8642	0,8809	0,8836	0,9324
word2vec (CBOW)	200	200	0,8611	0,8656	0,8809	0,8857	0,9344
word2vec (CBOW)	300	100	0,8619	0,8695	0,8823	0,8899	0,9372
word2vec (CBOW)	300	300	0,8638	0,8756	0,8825	0,8899	0,9393
fastText (Skip-gram)	200	100	0,8477	0,8564	0,8775	0,8781	0,9271
fastText (Skip-gram)	200	200	0,8513	0,8588	0,8789	0,8798	0,9280
fastText (Skip-gram)	300	100	0,8571	0,8590	0,8804	0,8806	0,9295
fastText (Skip-gram)	300	300	0,8583	0,8617	0,8806	0,8817	0,9315
fastText (CBOW)	200	100	0,8440	0,8528	0,8584	0,8683	0,9204
fastText (CBOW)	200	200	0,8442	0,8537	0,8689	0,8713	0,9237
fastText (CBOW)	300	100	0,8455	0,8552	0,8695	0,8718	0,9257
fastText (CBOW)	300	300	0,8465	0,8556	0,8741	0,8730	0,9265
GloVe	200	100	0,8315	0,8435	0,8451	0,8499	0,8986
GloVe	200	200	0,8321	0,8439	0,8474	0,8588	0,9074
GloVe	300	100	0,8393	0,8466	0,8526	0,8626	0,9138
GloVe	300	300	0,8414	0,8489	0,8565	0,8643	0,9192

architecture achieved the best prediction performance on the dataset, with a predictive performance of 93.7777%. The empirical results show that the second highest prediction performance is achieved by the CGWELSTM-v3 architecture with a classification accuracy of 90.9731%. In the CGWELSTM-v3 architecture, a bidirectional gated repetitive unit (bi-GRU)

Table 5. F-measure of the conventional architectures and the proposed scheme.

Word Embedding	Vector size	Dimension of projection layer	CNN	RNN	LSTM	GRU	CGWELSTM
word2vec (Skip-gram)	200	100	0,8620	0,8744	0,8784	0,8860	0,9374
word2vec (Skip-gram)	200	200	0,8661	0,8746	0,8829	0,8861	0,9399
word2vec (Skip-gram)	300	100	0,8684	0,8816	0,8842	0,8904	0,9407
word2vec (Skip-gram)	300	300	0,8718	0,8871	0,8994	0,8925	0,9520
word2vec (CBOW)	200	100	0,8542	0,8598	0,8764	0,8791	0,9277
word2vec (CBOW)	200	200	0,8567	0,8612	0,8765	0,8812	0,9297
word2vec (CBOW)	300	100	0,8576	0,8651	0,8778	0,8853	0,9325
word2vec (CBOW)	300	300	0,8595	0,8712	0,8781	0,8854	0,9345
fastText (Skip-gram)	200	100	0,8434	0,8520	0,8731	0,8737	0,9224
fastText (Skip-gram)	200	200	0,8470	0,8544	0,8744	0,8753	0,9233
fastText (Skip-gram)	300	100	0,8528	0,8546	0,8759	0,8761	0,9248
fastText (Skip-gram)	300	300	0,8540	0,8573	0,8761	0,8773	0,9268
fastText (CBOW)	200	100	0,8397	0,8485	0,8541	0,8639	0,9158
fastText (CBOW)	200	200	0,8399	0,8493	0,8645	0,8668	0,9190
fastText (CBOW)	300	100	0,8412	0,8509	0,8651	0,8674	0,9210
fastText (CBOW)	300	300	0,8422	0,8513	0,8697	0,8686	0,9218
GloVe	200	100	0,8273	0,8393	0,8408	0,8456	0,8941
GloVe	200	200	0,8279	0,8396	0,8431	0,8545	0,9028
GloVe	300	100	0,8350	0,8423	0,8483	0,8582	0,9092
GloVe	300	300	0,8371	0,8446	0,8521	0,8599	0,9145
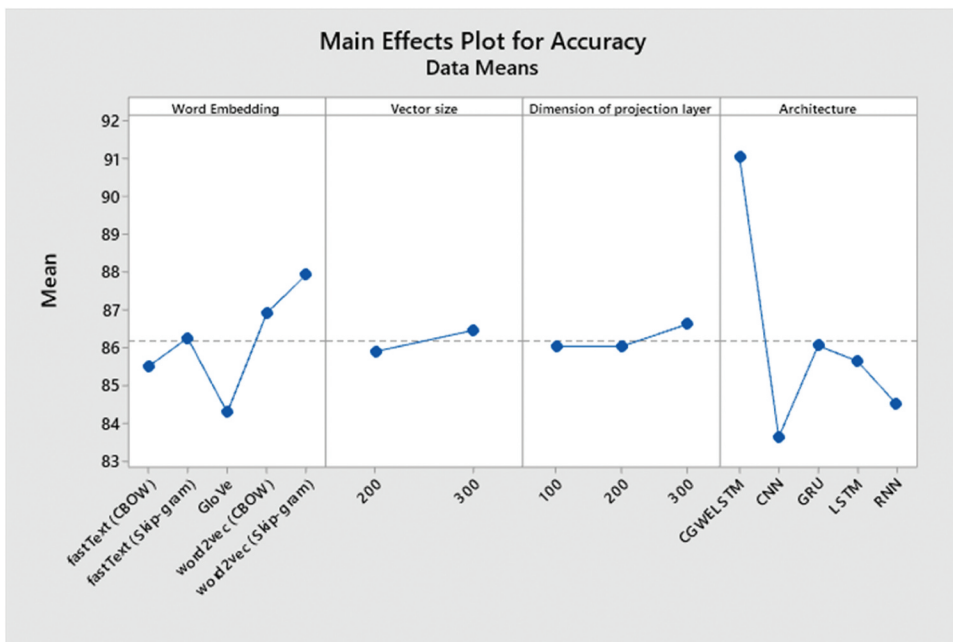
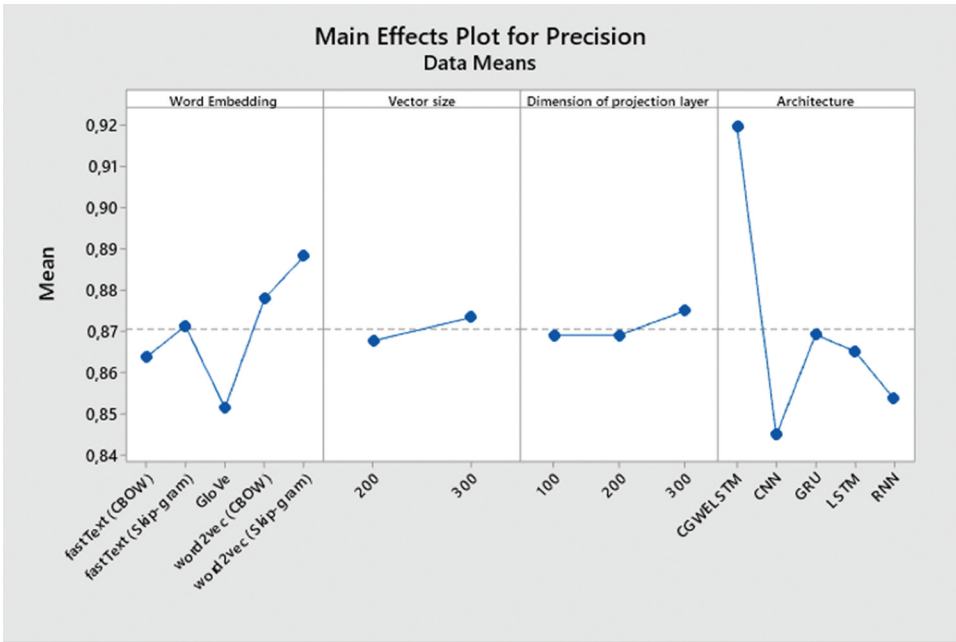**Figure 2.** The summary figure for accuracy.

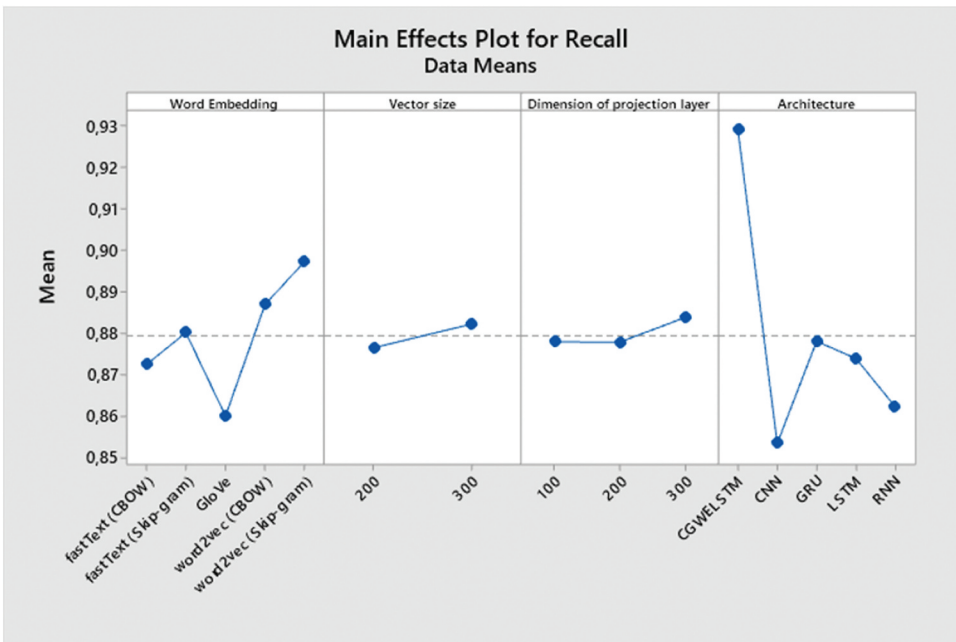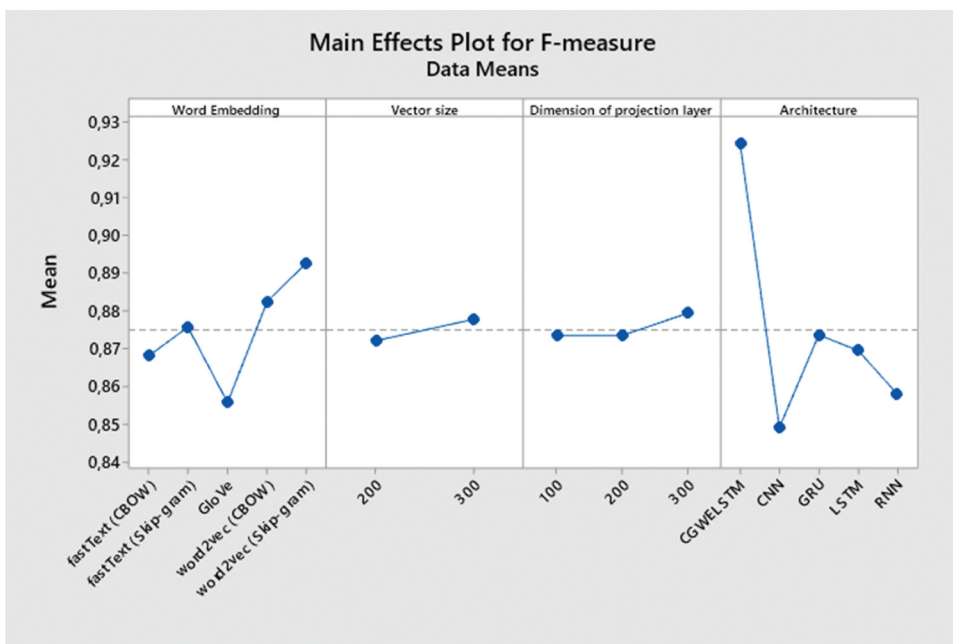**Figure 3.** The summary figure for precision.

Figure 4. The summary figure for recall.

Figure 5. The summary figure for F-measure.

is used in the bidirectional layer instead of bidirectional long short-term memory (bi-LSTM). Therefore, the empirical results show that bi-LSTM outperforms bi-GRU for text classification tasks in benchmarks. The third highest predictive performance was achieved by the ABCDM architecture, followed by the CAT-BiGRU architecture. For the results listed in Table 6, tree-LSTM, tree-bi-LSTM and SS-BED architectures give lower prediction performances on the dataset. According to the classification accuracies listed in Table 5, the use of CNN with RNN generally outperforms traditional deep neural network architectures. In addition, bidirectional architectures often achieve promising results.

The precision, recall, and F-measure metrics from the deep neural network models are also shown in Table 6. For the dataset employed in the empirical investigation, CGWELSTM outperformed the other architectures in terms of precision, recall, and F-measures. With a precision of 0.9472, recall of 0.9569, and F-measurement value of 0.9520, CGWELSTM achieved the best prediction performance in the dataset.

Three model variations were taken to evaluate the performance of the CGWELSTM. As can be seen from the results in Table 6, the use of groupwise improvement can yield better performance. In addition, bi-LSTM outperforms bi-GRU and the utilization of attention mechanisms can yield better predictive performance.

Table 6. Predictive performances by the deep learning models.

Architecture	Accuracy	Precision	Recall	F-measure
Tree-LSTM	82,8389	0,8359	0,8376	0,8368
Tree bi-LSTM	83,3402	0,8410	0,8427	0,8418
SS-BED	83,5993	0,8436	0,8453	0,8444
HAN	85,7452	0,8652	0,8670	0,8661
AGCNN	85,9673	0,8675	0,8692	0,8684
CNN-GRU	86,1370	0,8692	0,8709	0,8701
AC-BiLSTM	86,1814	0,8696	0,8714	0,8705
WCNNLSTM	86,1943	0,8698	0,8715	0,8706
IWV	86,5722	0,8736	0,8754	0,8745
ARC	87,6503	0,8845	0,8863	0,8854
ATTPooling	88,4448	0,8925	0,8943	0,8934
CGWELSTM-v2	88,5043	0,8930	0,8948	0,8939
CRNN	88,5258	0,8933	0,8951	0,8942
CGWELSTM-v1	88,8956	0,8979	0,9021	0,9000
CAT-BiGRU	89,3312	0,9014	0,9032	0,9023
ABCDM	90,1395	0,9096	0,9114	0,9105
CGWELSTM-v3	90,9731	0,9189	0,9283	0,9236
CGWELSTM	93,7737	0,9472	0,9569	0,9520

In Table 7, the predictive performance values obtained by the deep neural learning models on Sentiment-140 dataset has been presented. The empirical results indicate that the proposed scheme (i.e., CGWELSTM architecture) outperforms the state-of-the-art architectures on Sentiment-140 dataset. CGWELSTM architecture achieved the best predictive performance on the dataset, with a classification accuracy of 85.4001%. The empirical results indicate that the second highest predictive performance has been achieved by CGWELSTM-v3 architecture, with a classification accuracy of 83.9217%. In the CGWELSTM-v3 architecture, bidirectional gated recurrent unit (bi-GRU) has been employed in the bidirectional layer, instead of bidirectional long

Table 7. Predictive performances by the deep learning models on Sentiment-140 dataset.

Architecture	Accuracy	Precision	Recall	F-measure
Tree-LSTM	77,1687	0,7803	0,7779	0,7791
Tree bi-LSTM	77,1807	0,7804	0,7780	0,7792
SS-BED	77,2500	0,7811	0,7787	0,7799
HAN	77,5033	0,7837	0,7813	0,7825
AGCNN	78,1518	0,7902	0,7878	0,7890
CNN-GRU	78,7938	0,7967	0,7943	0,7955
AC-BiLSTM	79,2335	0,8011	0,7987	0,7999
WCNNLSTM	79,5461	0,8043	0,8019	0,8031
IWV	80,1607	0,8105	0,8081	0,8093
ARC	80,8119	0,8171	0,8146	0,8159
ATTPooling	81,2609	0,8216	0,8192	0,8204
CGWELSTM-v2	81,3622	0,8218	0,8202	0,8210
CRNN	81,4044	0,8231	0,8206	0,8219
CGWELSTM-v1	78,946	0,8274	0,8256	0,8265
CAT-BiGRU	81,9217	0,8283	0,8258	0,8271
ABCDM	82,4631	0,8338	0,8313	0,8325
CGWELSTM-v3	83,1905	0,8403	0,8489	0,8446
CGWELSTM	85,4001	0,8626	0,8714	0,8670
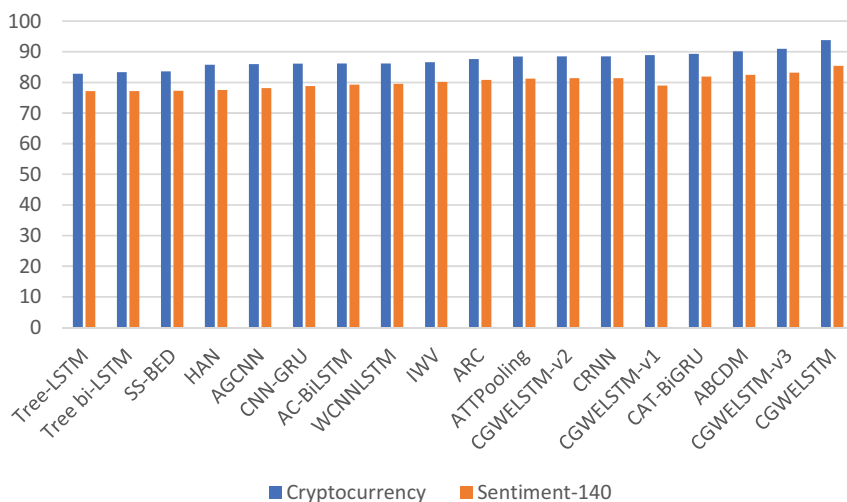
Figure 6. The bar-chart for the models based on classification accuracy.

short-term memory (bi-LSTM). Hence, the empirical results indicate that bi-LSTM outperforms bi-GRU for the text classification tasks on the dataset. The third highest predictive performance has been achieved by the ABCDM architecture, which is followed by CAT-BiGRU architecture. As it can be observed from the results listed in Table 7, the tree-LSTM, tree-bi-LSTM, and SS-BED architectures yield lower predictive performances on the datasets. According to the classification accuracy values listed in Table 6, the utilization of CNN in conjunction with RNN generally outperforms the conventional deep neural network architectures. In addition, the bidirectional architectures generally obtain promising results.

In Figure 6, the bar-chart for the models based on the classification accuracy values has been presented. As illustrated in Figure 6, the utilization of group-wise enhancement mechanism improves the predictive performance of the deep neural network architectures. Similarly, by utilizing the attention mechanism, the predictive performance of deep neural network architectures is enhanced. Regarding the predictive performance of RNN-based architectures utilized in the bidirectional layer, the proposed scheme in which bi-LSTM has been utilized outperforms the scheme with bi-GRU (referred as, CGWELSTM-v3 architecture).

Conclusion

Sentiment analysis (SA) is a rapidly growing branch of natural language processing research. Deep architectures have been widely used in SA due to their high predictive efficiency. RNNs are particularly adept at simulating long-term dependencies and extracting global features. Convolutional neural

networks (CNNs) are capable of successfully extracting local features. Hybrid architectures combining RNN, and CNN have been shown to outperform traditional deep neural network architectures. We present a bidirectional CNN-RNN architecture for text sentiment classification in this article, as well as group-wise enhancement and attention mechanisms. Local features are derived using a convolution layer, and the weight values associated with insightful features are enhanced by the proposed scheme. The attention mechanism and the fully connected layer were used after feeding the improved context vector to the bidirectional layer to capture global features. The experimental results indicate that when performing SA tasks, the proposed scheme outperforms state-of-the-art architectures. The empirical results indicate that the proposed scheme (i.e., CGWELSTM architecture) outperforms the state-of-the-art architectures on the dataset taken into consideration in the empirical analysis. CGWELSTM architecture achieved the best predictive performance on the dataset, with a classification accuracy of 93.7737%.

Disclosure statement

No potential conflict of interest was reported by the author(s).

ORCID

Mansur Alp Toçoğlu ⬤ http://orcid.org/0000-0003-1784-9003

References

Abraham, J., D. Higdon, J. Nelson, and J. Ibarra. 2018. Cryptocurrency price prediction using Tweet volumes and sentiment analysis. *SMU Data Science Review* 1 (3):1–21.

Aslam, N., F. Rustam, E. Lee, P. B. Washington, and I. Ashraf. 2022. Sentiment analysis and emotion detection on cryptocurrency related tweets using ensemble LSTM-GRU model. *IEEE Access* 10:39313–24. doi:10.1109/ACCESS.2022.3165621.

Basiri, M. E., S. Nemati, M. Abdar, E. Cambria, and U. R. Acharya. 2021. ABCDM: An attention-based bidirectional CNN-RNN deep model for sentiment analysis. *Future Generation Computer Systems* 115:279–94. doi:10.1016/j.future.2020.08.005.

Calefato, F., F. Lanubile, and N. Novielli. 2017. EmoTxt: A toolkit for emotion recognition from text. Proceedings of the 7th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos, 79–80, San Antonio, TX, USA, 2017 October.

Chatterjee, A., U. Gupta, M. K. Chinnakotla, R. Srikanth, M. Galley, and P. Agrawal. 2019. Understanding emotions in text using deep learning and big data. *Computers in Human Behavior* 93:309–17. doi:10.1016/j.chb.2018.12.029.

Cho, K., B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation *ArXiv* 2014: 1–15 . .

Chuen, D. L. K., L. Guo, and Y. Wang. 2017. Cryptocurrency: A new investment opportunity? *The Journal of Alternative Investments* 20 (3):16–40. doi:10.3905/jai.2018.20.3.016.

Chung, J., C. Gulcehre, K. Cho, and Y. Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling *ArXiv* 2014: 1–9.

Colianni, S., S. Rosales, and M. Signorotti. 2015. Algorithmic trading of cryptocurrency based on twitter sentiment analysis. *CS229 Project, Stanford University* 1 (5):1–4.

Çılgın, C., C. Ünal, S. Alıcı, E. Akkol, and Y. Gökşen. 2020. Metin Sınıflandırmada Yapay Sinir Ağları ile Bitcoin Fiyatları ve Sosyal Medyadaki Beklentilerin Analizi. *Mehmet Akif Ersoy Üniversitesi Uygulamalı Bilimler Dergisi* 4 (1):106–26. doi:10.31200/makuubd.651904.

Di Pierro, M. 2017. What is the blockchain? *Computing in Science & Engineering* 19 (5):92–95. doi:10.1109/MCSE.2017.3421554.

Go, A., R. Bhayani, and L. Huang. 2009. Twitter sentiment classification using distant supervision. *CS224N Project, Stanford University* 1 (12):1–6.

Gutiérrez, G., J. Canul-Reich, A. O. Zezzatti, L. Margain, and J. Ponce. 2018. Mining: Students comments about teacher performance assessment using machine learning algorithms. *International Journal of Combinatorial Optimization Problems and Informatics* 9 (3):26–40.

Haryadi, D., and G. P. Kusuma. 2019. Emotion detection in text using nested long short-term memory. *International Journal of Advanced Computer Science and Applications* 10 (6):351–57. doi:10.14569/IJACSA.2019.0100645.

Hasan, M., E. Rundensteiner, and E. Agu. 2019. Automatic emotion detection in text streams by analyzing twitter data. *International Journal of Data Science and Analytics* 7 (1):35–51. doi:10.1007/s41060-018-0096-z.

Hochreiter, S., and J. Schmidhuber. 1997. Long short-term memory. *Neural Computation* 9 (8):1735–80. doi:10.1162/neco.1997.9.8.1735.

Huang, X., W. Zhang, X. Tang, M. Zhang, J. Surbiryala, V. Iosifidis, Z. Liu, and J. Zhang. 2021. LSTM based sentiment analysis for cryptocurrency prediction *ArXiv* 2021: 1–4.

Kamal, A., and M. Abulaish. 2022. CAT-BiGRU: Convolution and attention with bi-directional gated recurrent unit for self-deprecating sarcasm detection. *Cognitive computation* 14 (1):91–109. doi:10.1007/s12559-021-09821-0.

Köksal, B., G. Erdem, C. Türkeli, and Z. K. Öztürk. 2021. Twitter'da Duygu Analizi Yöntemi Kullanılarak Bitcoin Değer Tahminlemesi. *Düzce Üniversitesi Bilim ve Teknoloji Dergisi* 9 (3):280–97. doi:10.29130/dubited.792909.

Lamon, C., E. Nielsen, and E. Redondo. 2017. Cryptocurrency price prediction using news and social media sentiment. *SMU Data Science Review* 1 (3):1–22.

Li, X., X. Hu, and J. Yang. 2019. Spatial group-wise enhance: Improving semantic feature learning in convolutional networks. *ArXiv Preprint ArXiv:1905.09646*.

Li, J., M. T. Luong, D. Jurafsky, and E. Hovy. 2015. When are tree structures necessary for deep learning of representations? *ArXiv* Preprint *ArXiv*:1503.00185.

Liu, G., and J. Guo. 2019. Bidirectional LSTM with attention mechanism and convolutional layer for text classification. *Neurocomputing* 337:325–38. doi:10.1016/j.neucom.2019.01.078.

Liu, Y., L. Ji, R. Huang, T. Ming, C. Gao, and J. Zhang. 2019. An attention-gated convolutional neural network for sentence classification. *Intelligent Data Analysis* 23 (5):1091–107. doi:10.3233/IDA-184311.

Li, X., and X. Wu. 2015. Constructing long short-term memory based deep recurrent neural networks for large vocabulary speech recognition. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, 4520–24, South Brisbane, QLD, Australia, 2015 April.

Mehta, T., G. Kolase, V. Tekade, R. Sathe, and A. Dhawale. 2020. Price prediction and analysis of financial markets based on news social feed and sentiment index using machine learning

and market data. *International Research Journal of Engineering and Technology Technology* 7 (6):483–89.

Mikolov, T., I. Sutskever, K. Chen, G. Corrado, and J. Dean. 2013. Distributed representations of words and phrases and their compositionality Advances in Neural Information Processing Systems 26 3111–3119.

Onan, A. 2021. Sentiment analysis on product reviews based on weighted word embeddings and deep neural networks. *Concurrency and Computation: Practice and Experience* 33 (23): e5909. doi:10.1002/cpe.5909.

Onan, A. 2022. Bidirectional convolutional recurrent neural network architecture with group-wise enhancement mechanism for text sentiment classification. *Journal of King Saud University-Computer and Information Sciences* 34 (5):2098–117. doi:10.1016/j.jksuci.2022.02.025.

Onan, A., and M. A. Toçoğlu. 2021. Weighted word embeddings and clustering-based identification of question topics in MOOC discussion forum posts. *Computer Applications in Engineering Education* 29 (4):675–89. doi:10.1002/cae.22252.

Pant, D. R., P. Neupane, A. Poudel, A. K. Pokhrel, and B. K. Lama. 2018. Recurrent neural network based bitcoin price prediction by twitter sentiment analysis. Proceedings of the IEEE 3rd International Conference on Computing, Communication and Security, Kathmandu, Nepal, 128–32, 2018 October.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, and B. Thirion. 2011. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research* 12:2825–30.

Raju, S. M., and A. M. Tarif. 2020. Real-time prediction of BITCOIN price using machine learning techniques and public sentiment analysis *ArXiv* 2020 1–14.

Rasool, A., Q. Jiang, Q. Qu, and C. Ji. 2021. WRS: A novel word-embedding method for real-time sentiment with integrated LSTM-CNN model. Proceedings of the 2021 IEEE International Conference on Real-time Computing and Robotics, 590–95, Xining, China, 2021 July.

Rezaeinia, S. M., R. Rahmani, A. Ghodsi, and H. Veisi. 2019. Sentiment analysis based on improved pre-trained word embeddings. *Expert Systems with Applications* 117:139–47. doi:10.1016/j.eswa.2018.08.044.

Rojas-barahona, L. M. 2016. Deep learning for sentiment analysis. *Language and Linguistics Compass* 10 (12):701–19. doi:10.1111/lnc3.12228.

Salam, S. A., and R. Gupta. 2018. Emotion detection and recognition from text using machine learning. *International Journal of Computer Science and Engineering* 6 (6):341–45. doi:10.26438/ijcse/v6i6.341345.

Shah, F. M., A. S. Reyadh, A. I. Shaafi, S. Ahmed, and F. T. Sithil. 2019. "Emotion detection from tweets using AIT-2018 dataset." *Proceedings of the 5th International Conference on Advances in Electrical Engineering, Dhaka, Bangladesh*, 2019 September: 575–80.

Şaşmaz, E., and F. B. Tek. 2021. Tweet sentiment analysis for cryptocurrencies. Proceedings of the 6th International Conference on Computer Science and Engineering, 613–18, Ankara, Turkey, 2021 September.

Tikhomirov, S., E. Voskresenskaya, I. Ivanitskiy, R. Takhaviev, E. Marchenko, and Y. Alexandrov. 2018. Smartcheck: Static analysis of ethereum smart contracts. Proceedings of the 1st International Workshop on Emerging Trends in Software Engineering for Blockchain, Gothenburg, Sweden 2018 May: 9–16.

Tocoglu, M. A., O. Ozturkmenoglu, and A. Alpkocak. 2019. Emotion analysis from Turkish tweets using deep neural networks. *IEEE Access* 7:183061–69. doi:10.1109/ACCESS.2019.2960113.

Usama, M., B. Ahmad, E. Song, M. S. Hossain, M. Alrashoud, and G. Muhammad. 2020. Attention-based sentiment analysis using convolutional and recurrent neural network. *Future Generation Computer Systems* 113:571–78. doi:10.1016/j.future.2020.07.022.

Valencia, F., A. Gómez-Espinosa, and B. Valdés-Aguirre. 2019. Price movement prediction of cryptocurrencies using sentiment analysis and machine learning. *Entropy* 21 (6):589. doi:10.3390/e21060589.

Wang, J., L. C. Yu, K. R. Lai, and X. Zhang. 2016. "Dimensional sentiment analysis using a regional CNN-LSTM model." Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany 2016 August: 225–30.

Wen, S., and J. Li. 2018. "Recurrent convolutional neural network with attention for twitter and yelp sentiment classification: ARC model for sentiment classification." *Proceedings of the 2018 International Conference on Algorithms, Computing and Artificial Intelligence, Sanya, Hainan, China* 2018 December: 1–7.

Wołk, K. 2020. Advanced social media sentiment analysis for short-term cryptocurrency price prediction. *Expert Systems* 37 (2):e12493. doi:10.1111/exsy.12493.

Wöhrer, M., and U. Zdun. 2018. "Design patterns for smart contracts in the ethereum ecosystem." Proceedings of the 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data, Halifax, NS, Canada 2018 July:1513–20.

Yang, Z., D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy. 2016. Hierarchical Attention networks for document classification. Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego, CA 2016 June: 1480–89.

Zhang, X., W. Li, H. Ying, F. Li, S. Tang, and S. Lu. 2020. Emotion detection in online social networks: a multilabel learning approach. *IEEE Internet of Things Journal* 7 (9):8133–43. doi:10.1109/JIOT.2020.3004376.

Zhu, X., P. Sobhani, and H. Guo. 2015. Long short-term memory over tree structures. *ArXiv Preprint ArXiv*:1503.04881.