

Deep Learning Approach for Emotion Recognition Analysis in Text Streams

Changxiu Liu, School of Foreign Language, Guizhou University of Finance and Economics, China*

S. Kirubakaran, Jayamukhi Institute of Technological Sciences, India

Alfred Daniel J., SNS College of Technology, India

ABSTRACT

Social media sites employ various approaches to track feelings, including diagnosing neurological problems, including fear, in people or assessing a population public sentiment. One essential obstacle for automatic emotion recognition principles is variable with fluctuating limitations, language, and interpretation shifts. Therefore, in this paper, a deep learning-based emotion recognition (DL-EM) system has been proposed to describe the various relational effects in emotional groups. A soft classification method is suggested to quantify the tendency and allocate a message to each emotional class. A supervised framework for emotions in text streaming messages is developed and tested. Two of the major activities are offline teaching assignments and interactive emotion classification techniques. The first challenge offers templates in text responses to describe sentiment. The second activity includes implementing a two-stage framework to identify live broadcasts of text messages for dedicated emotion monitoring.

KEYWORDS

Accuracy, Classification, Deep Learning, Emotion Recognition, Text Streams

1. INTRODUCTION

Emotion recognition (El Hammoumi et al, 2018) is the process of identifying human emotion. Generally, people differ in their precision at the recognition of the feelings of others. The use of technology to assist people with emotion detection (Yoon et al, 2019) is a relatively growing area. Currently, most research has been performed on automating the recognition of facial expression from the video (Manogaran et al, 2019), spoken word from audio (Ahmad et al, 2021), written expression from text and physiology expressions from wearable devices (Mukhopadhyay et al, 2020). Emotion identification (Alazab,2020) from text is a recent essential research area in the field of natural language processing (NLP) and deep learning (Mamoun Alazab et al, 2019), which may reveal some valuable input for a variety of purposes. Nowadays, a large volume of textual data has been produced in real-time due to communication technologies' development (Iqbal et al, 2019). Some of the applications are social media posts, micro-blogs, news articles, online teaching and assessment, customer reviews, etc. (Kwon et al, 2016).

DOI: 10.4018/IJTHI.313927

*Corresponding Author

This article published as an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

This short-text content can be a valuable resource for text mining (Elmasry et al, 2018) to uncover different aspects such as emotions, behaviours (Muthu et al, 2020), and so on. E-learning (Yi et al, 2020) is one of the most essential and useful teaching environments for supporting and understanding student learning issues during the pandemic time. (Chen et al, 2014; Tai et al, 2012) Nowadays, students can interact with teachers or other students using lots of collaborative e-learning tools such as Google classrooms, Loop, Edmodo, Storybird, Packet, etc. However, teachers often remain outside of this process and do not understand the blended learning environment's actual difficulties (Young et al, 2018; Al-Turjman et al, 2019). In the collaborative teaching environment (Ullah et al, 2019; Tandel et al, 2019), sentimental or emotional factors affect student motivation and learning outcomes during online or offline classes. Detecting and managing emotions (Jacob et al, 2019) is an essential learning activity; it contributes to motivating students to improve learning outcomes. In the field of human-computer interaction (Ullah et al, 2019), it plays a vital role in emotion detection using text analysis, image analysis (Hulliyah et al, 2019), and speech signals analysis (Clarizia et al, 2018). The text mining-based emotion detecting approach is used in many real-world problems, including teaching material or feedback in e-learning, customer emotions-based product recommendation, monitoring customer satisfaction, etc. (Serte et al, 2020). This research developed an automatic extraction of students' emotions from online text streaming(Ullah et al, 2018) during online classes. (El Hammoumi et al, 2018)This research's basic idea is to create a non-disturbing method to identify students' emotions from text streaming during online teaching and assessment to offer feedback or motivation. (Rajalingam et al, 2020)The research has been adapted a deep learning-based text analysis method for students' emotion detection using various NLP steps. The deep model contains multiple stages such as text data pre-processing, feature extraction from the text, feature selection from that extracted emotion feature set, and a Deep Fish Swarm Optimized Gated Recurrent Unit Neural Network(Deep FSOGRUNN) model detecting various emotions of students(Nasim et al, 2017 ; Ghanbari et al, 2017). The sources of streaming text data can be transformed into valuable analysis results when handling these methods (Jafar et al, 2019). The system's main objective is to monitor various emotions by retrieving students' text responses from the Chabot and sending feedback messages to each emotion group to improve the learning process (Chang-Hyun Park et al, 2003). The research also contributes to improving the accuracy of a deep learning approach in text emotion detection (Hakdağlı et al, 2018).

Emotional recognition is a dynamic process which focuses on the person's emotional status, meaning the emotions which correlate to the acts of each person which are distinct. In general, human people transmit their sentiments in many ways. Here, a correct perception of these emotions is vital to promote meaningful conversation. Emotional awareness is vital for social interactions nonetheless in our everyday lives, and emotions play a part in deciding on human activity.

A technique to identify the various relationship impacts in emotional groups has been presented. To quantify the propensity and provide an emotional class message, a soft classification approach is proposed. A supervised emotional framework is created and tested for text streaming communications. Offline training and interactive classification approaches are two of the key activities. The first task provides templates for describing the feeling in-text answers. The second activity consists of the implementation of a two-stage framework to recognize live text message transmissions for the monitoring of emotion.

The research's organization work in the subsequent order; section 1 discusses the introduction and research objective. Section 2 discusses the recent related research on emotion detection and various methods utilized in this research. Section 3 and 4 discuss the methodologies and the evaluation results of the automatic emotion detection system. Finally, section 5 discusses the conclusion of this research findings and feature researches as well. The consequent section describes the literature review in detail.

2. LITERATURE REVIEW

This section discusses the various studies related to text based on different human emotions or sentiment detection approaches and multiple studies on text mining techniques.

This study (Yadollahi et al, 2017) prepared a survey to clarify the link between emotion and opinion mining. It discusses polarity classification, dataset resources, emotion and opinion mining techniques and suggests various text-based sentiment analysis steps. This study (Hajah et al, 2020) developed a text-document transformation approach using Laplace smoothing technique in naive bayes vectors. The smoothing method assures posterior probabilities values never become zero, and the logarithmic function ensures that probability calculations become very small. The technique attained the maximum of 99% of accuracy in converting text to numeric conversion.

In (Sailunaz et al, 2019), emotion detection and sentiment analysis has been developed using Twitter posts to generate a personalized recommendation system and measure users' influential scores from tweets and replies. ISEAR, SemEval, EmoBank, and TREC datasets are used to evaluate system performance. In (Batbaatar et al, 2019), Bidirectional long short-term memory (Bi-LSTM) and convolution neural network (CNN) models have been combined to analyze the semantic emotions of text data. The CNN model is utilized to extract text features to train the Bi-LSTM model. Moreover, the analysis process is performed with ten different texts, emotion-related datasets. This study (Al-Omari et al, 2020), deep, dense model, is combined with Bi-LSTM to improve the English language's emotion detection efficiency. Moreover, the Bidirectional Encoder Representation from Transformer (BERT) embedding, Glove word embeddings, and psycholinguistic feature-based methods extract the text emotion feature.

Two alternative forms of writing may be presented: formal and casual writing. Text, either online or offline. A writing piece might include a lot of sentiments, emotions and thoughts. In opinion mining and sentiment analysis, several strategies and strategies are present to extract texts' emotions. The current article proposes a differential analysis in the realm of sentiment classification of formal and informal texts. This article includes research and analysis of variations in methodologies used to detect emotions and to analyze sentiments for both scenarios.

Table 1 contains the various deep model-based text emotion detection-related research. This study (Majumder et al, 2017) introduced a tailored transfer learning-based decision support system for individuals' emotion states. It used the Bi-LSTM model to detect six different dataset's emotional conditions and obtained 58.6% accuracy.

This study (Chatterjee et al, 2019) investigated the efficiency of deep LSTM model in textural emotion detection for six different emotional groups. The evolution results show that the LSTM model is obtained maximum efficiency (94.7%). This study (Abdullah et al, 2018) mainly focuses on the pre-processing phase in text mining to improve accuracy. It utilizes the Term Frequency Inverse Document Frequency (TFIDF) approach to extract text features from documents, Latent semantic analysis (LSA) and linear discriminate analysis (LDA) for feature selection. The CNN model is attained by 91% for TFIDF. The model (Mohaouchane et al, 2019) used an attention mechanism to the words that are important to identify sentiments. The attention mechanism is combined with CNN to extract the local features and bidirectional gated recurrent unit (Bi-GRU) neural network for sentiment classifications. The model has been obtained a maximum of 92.90% accuracy. This research (FaraShatnawi et al, 2020), and improved artificial fish swarm optimizer (FSO), is introduced for weapon target assignment problems in the air defense system to improve the assignment rate. The individual visual of artificial fish and genetic operator in PSO is incorporated to avoid the maximum local trap. This study (Zobeidi et al, 2017) introduces a filtering based feature ranking and selection approach to reduce dimensionality. This approach combines and uses three methods such as information gain, chi-square statistic and inter-correlation. This approach stabilizes each feature score and gives true ranking. Therefore, it increases the stability of the variable's scores without losing the overall accuracy.

Table 1. Summary of related researches on text-based various emotions detection using deep models

Ref	Dataset	Research Problem	Methods	Accuracy
(Ullah et al, 2019)	DailyDialog, Crowdflower, TEC, Tales-Emotions, ISEAR, EmoInt, Electoral Tweets, Grounded-Emotions, Emotion cause, and SemEval 2016 (Sentence sentiment emotions SSEC)	Semantic Emotion analysis	Bi-LSTM+CNN	F1-98.8%
(Hulliyah et al, 2017)	EmoContext of semeval 2019 contains Happy 4243, sad 5463, angry 5506, and others 14948	Emotion detection in the English language	(BERT) +Bi-LSTM	91.99%
(Clarizia et al, 2018)	James Pannebaker and Laura King’s stream of consciousness essay dataset	Personality detection	CNN	62.68%
(Serte et al, 2020)	Twitter Firehose (annotated labels, Happy 109, sad 107, angry 90, and others 1920 (17.62 million tweet conversations)	Sentiment semantics-based emotion understanding	LSTM	F1-80.79%
(Ullah et al, 2020)	SemEval-2018(Arabic tweets) contains both training and test data of Joy 952+448, angry 1027+373, sad 1030+370, fear 1028+372, and sentiment 1070+730	Sentiment and emotion in Arabic tweets	LSTM+ CNN	81.8%
(El Hammoui et al, 2018)	Arabic 15050Youtube comments	Detecting offensive language on Arabic social media	CNN+Bi-LSTM	83.46%
(Rajalingam et al, 2020)	Task 7 of SemEval2020, (humicro-edit dataset news headline 15095, 14696)	Humour detection in news	(BERT) + regression	62.291%
(Nasim et al, 2017)	Reuters 21578 and 20 newsgroup dataset	Text mining	Fuzzy min-max neural network (FMMNN) and principal component analysis	95%

Social media use several ways to measure feelings, including the diagnosis of neurological issues, fear in individuals or assessment of public mood. The changeable limits, language and interpretation are one key barrier for automatic principles of emotional recognition.

To quantify the propensity and provide an emotional class message, a soft classification approach is proposed. A supervised emotional framework is created and tested for text streaming communications. Offline training and interactive classification approaches are two of the key activities. The first task provides templates for describing the feeling in-text answers. The second activity consists of implementing a two-stage framework to recognise live text message transmissions for the monitoring of emotion. Also, DL-EM uses an online means of assessing collective emotion and identifying incidents in live text sources.

Most of the existing papers mainly focused on word embedding vectors representing rich semantic or systematic data and not covered the emotional relationship between words. Some research covers the research gap but obtaining low accuracy in finding emotion groups. In this research, an deep learning approach has been utilized to resolve this research gap in previous research. The deep learning models

analyze students' text-streaming comments to automatically identify each student's emotional state and send learning feedback to each student's groups separately. The deep model contains various stages such as text data pre-processing, feature extraction from the text, feature selection from that extracted emotion feature set, and finally, a deep classifier detecting students' various emotions. The subsequent section discusses the methodologies of the emotion detection models.

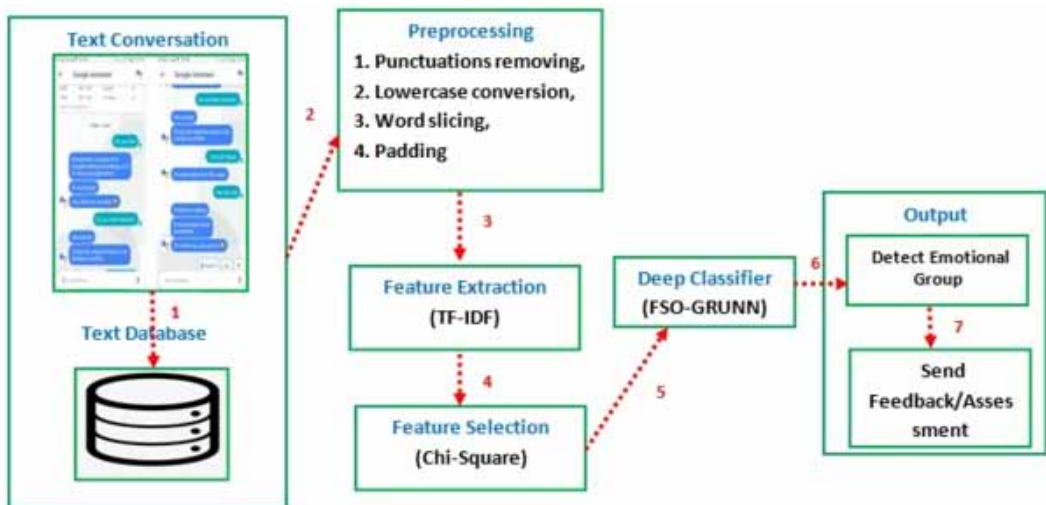
3. METHODOLOGY OF EMOTION RECOGNITION ANALYSIS IN TEXT STREAMS

Figure 1 illustrates the step by step conversation analysis process for emotional group detection. Initially, the input text conversations are documented and stored in a database. Next, the raw text dataset information has been pre-processed to remove unwanted textual information using various NLP pre-processing steps such as punctuations removing, lower case, word slicing, and padding. Next, the text features are extracted from the pre-processed text data using tf-idf extractor method. The extracted text-based feature dimensions are reduced by using chi-square based ranking method. Finally, the selected features are taken as input to the Deep FSOGRUNN model to detect various emotional groups and send feedback or assessment. The section discusses the data sources and methodologies of emotion recognition approaches.

3.1 Dataset Details

The Deep FSOGRUNN efficiency in text emotion detection has been trained and evaluated with six publicly available text emotion-related datasets. EmoInt (A)(49): WASSA-2017 Shared Task on Emotion Intensity, the dataset contains social media contents. The tweets are labelled via crowdsourcing with various emotions such as anger-1621, joy-1548, fear-2149, sadness—1458 (totally-6776). EmoBank (B) (50),(Sven Buechel et al, 2017):The dataset contains 10k sentences labelled with valence, arousal, and dominance values.The valence and arousal Facebook posts by preotiuc-pietro and others (C)(Chatterjee et al, 2018): The dataset contains annotated four types of anonymsmessages such as valence 1, valence 2, arousal 1, and arousal 2. The labeled intensities have 2895 messages. The emotions in text data set by crowded flower(D) (53): The dataset annotated emotions with intensities as joy-8142, fear-7799, sadness-4697, anger-1311, and disgust-164 (totally-22113). ISEAR (E)(54): The dataset contains five types of annotated emotions with intensities values, joy-1064, fear-1051,

Figure 1. General workflow of the Emotion recognition analysis in text streams



sadness-1012, anger-1058, and disgust-1056(Totally-5241). A real notation of the SemEval Stance data with emotions(F) (55): It contains annotated sentences in various emotions such as joy-232, fear-15, sadness-20, surprise-35, anger- 23 and disgust-10(Totally-312). As observed from Figure 1, that the step-by-step procedure of emotional group discovery conversation analysis. The text chats for the input are initially recorded and kept in a database. Next the information in the raw text dataset has been pre-processed to eliminate undesired textual information using several NLP pre-processing procedures, such as deleting punctuations, lower case, word slicing and padding. The text properties are then retrieved using the extractor technique from the pre-processed text data. By chi-square ranking algorithm, the retrieved text-based feature dimensions are minimised.

3.2 Pre-Processing

The collected text datasets are used to evaluate the efficiency of the text emotion detection system. The first step is to pre-process the given text data since it is essential to reduce the input data's unwanted information to improve the prediction accuracy.

In-text pre-processing unwanted text information is removed from the original text documents. The pre-processing contains three stages: punctuation removal (e.g. ?!,/,@,#,\$,%,&,*,(,),`~,), lower case transformation, and word slicing. Figure 2 illustrates three stages of pre-processing with example. The punctuations and special characters are removed from the text document; next, the punctuation-free sentence is converted into lower case. Finally, each word in a sentence is sliced with the help of space. Next, all the words in an expression are added with '0' padding to make equal (e.g. she00 said0 im000 very0 upset).

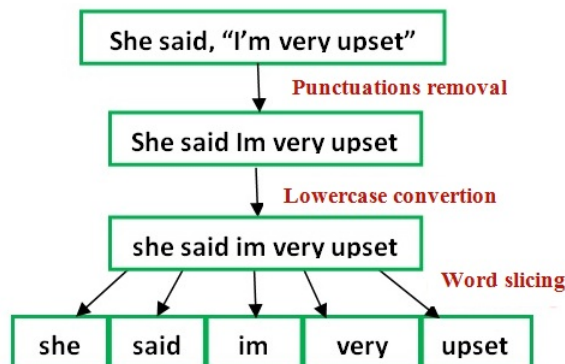
3.3 Feature Extraction

Feature extraction is one of the significant steps before text data analysis. In this emotion classification model, term frequency-inverse document frequency tf-idf method is adopted for feature extractions; it is a numerical statistic intended to reflect the word's importance to a document in a corpus. The tf-idf value increases proportionally to the amount of time a word presents in a document. It is offset by the number of documents in the collection that include the term used to adjust because some words give more repeatedly in common.

3.3.1 Term Frequency $tf(t, d)$

It is used to find a number of times t occurs in document d . The derivative for term frequency is given as follows:

Figure 2. Three-stage text data pre-processing



$$tf(t, d) = 0.5 + 0.5 \frac{f_{t,d}}{\max\{f_{t',d} : t' \in d\}} \quad (1)$$

In eqn(1), $f_{t,d}$ denotes the row frequency count, t indicates term or word occurrence in a document d . The augmented frequency is used to avoid a bias towards longer documents. It calculates the row frequency of the most occurring term in the document.

3.3.2 Inverse Document Frequency *idf*

It is used to find the importance of term or word, and the derivative of the logarithmically scaled inverse fraction of a document is derived as follows:

$$idf(t, D) = \log \frac{N}{|\{d \in D : t \in d\}|} \quad (2)$$

In eqn(2), the notation N denotes the total number of the document in the collections and $|\{d \in D : t \in d\}|$ denotes the number of the document where the word or term appears.

3.3.3 Term Frequency-Inverse Document Frequency *tf - idf*

$$tfidf(t, d, D) = tf(t, d) * idf(t, D) \quad (3)$$

A high frequency in eqn(3) is reached by a high term frequency and low document frequency weights; hence it filtered out common words. Since the ratio inside the log function always ≥ 1 , the value of $tf - idf \geq 0$, ratio inside $idf = 1$, carrying the *idf* and *tf-idf* closer to 0.

The *tf-idf* is created using information theory; the characteristic assumption of a probability distribution is:

$$p(d|t) = \frac{1}{|\{d \in D : t \in d\}|} \quad (4)$$

The eqn(4) denotes the assumption and implications of the heuristic that *tf-idf* is used:

$$M(T; DP) = \sum_{t,d} p_{t,d} * p_d * idf(t) = \sum_{t,d} tf(t, d) * \frac{1}{|D|} * idf(t) \quad (5)$$

$$M(T; DP) = \frac{1}{|D|} \sum_{t,d} tf(t, d) * \frac{1}{|D|} * idf(t) \quad (6)$$

The eqn(5) and eqn(6) shows the summing of all possible term and document recovers the mutual information between document and word.

Table 2 contains the sample tf-idf based feature vector for the given input: she said im very. The extracted text emotion feature value has been taken as input to the dimensionality reduction.

3.4 Dimensionality Reduction

The extracted high text features are normalized and reduced to improve classification performance. Information gain and chi-square-based score vectors are normalized using the min-max normalization techniques, denoted in eqn(7):

$$nv = \frac{av - \min_fv}{\max_fv - \min_fv} \left(n_{\max_rng} - n_{\min_rng} \right) + n_{\min_rng} \quad (7)$$

where nv is denoted as new normalized value, av represents the actual value of the feature igs_a and $chis_a$ vector. \min_fv and \max_fv minimum and maximum value of the feature vector and n_{\max_rng} and n_{\min_rng} denotes the maximum and minimum normalization range. The eqn(7) is normalized the input feature values between the range from 0 to 1.

The representation of the score vector is denoted as $sv_a = \begin{pmatrix} igs_a \\ chis_a \end{pmatrix}$ where the chi-square $chis_a$ score and information gain igs_a vector is taken to rank the score vector sv_a by calculating the magnitude of both score vector $|sv_a|$ by using the below eqn(8):

$$|sv_a| = \sqrt{(igs_a)^2 + (chis_a)^2} \quad (8)$$

The magnitude values are compared with the values of the features. If a text feature value is higher than the appropriate magnitude $|sv_a|$ value then it is considered as highest-ranked features. The selected significant parts are taken as the input to the FSOGRU network model. The functionalities of the classifier are explained in the subsequent section.

3.5 Deep Classification

The gated recurrent unit GRU network is an add-on version of long short- term memory (LSTM) network. The GRU network and LSTM work similar, but it doesn't use cell layer to transfer data. The GRU network has several benefits over LSTM, such as less expensive and faster performance. During the network train, the current input is learned from its previous hidden layer node's output. The minimal gated unit GU is similar to the fully connected gated unit. Still, this network is slightly

Table 2. Sample tf-idf based text feature extraction

Document 1	
Term	Term count
she	1
said	1
im	1
very	1
upset	1

different from fully connected GRU, which is the update and reset gate vector fused as forget gate. In this research, the FSOGRUNN has been utilized to identify students' similar emotion to send feedback. Using the gradient descent optimize the GRU network to update network weight by back-propagating the network, but this approach simple and consumes lots of time to predict the desired output. According to the earlier study, the GRU perform well for text-based emotion analysis; however, it gives more loss rate in identifying text emotions. This GRUNN model issue has been reduced by incorporating the FSO optimizer's objective function to improve the error rate. The FSO is one of the global optimization approaches, and it takes less computation time to reach global minima. The following derivations of the GRUNN classifier are used to detect the student's text emotions:

$$A = \begin{Bmatrix} A_{11} & A_{12} & \dots & A_{1j} \\ A_{21} & A_{22} & \dots & A_{2j} \\ \vdots & \vdots & \vdots & \vdots \\ A_{i1} & A_{i2} & \dots & A_{ij} \end{Bmatrix} \quad (9)$$

where in eqn(9) the A denotes the various text feature vector, which contains i number of text features attribute values and j number of text emotions features and A_{ij} denotes the position of input text feature value. The minimal GRU generally uses sigmoid and tangent activation functions to decide state activation for each node operations:

$$\begin{cases} for_t = \sigma_g(W_{for}A_t + U_{for}h_{t-1} + bi_{for}) \\ \sigma(ac) = (1 + e^{-pc})^{-1} \end{cases} \quad (10)$$

In eqn(10) the for_t denotes the forget gate value at time t and the symbol σ_g denotes the sigmoid activation function gate, ac indicates the actual text emotion class value, and pc indicate the predicted emotion class value. The default threshold range of the sigmoid activation is denoted as sigmoid (01), 0 represents an unsuccessful node, and 1 represents successful node but the values between >0 and <1 is considered for back-propagation. The $W_{for}GP_t$ denotes the weight of each forget gate nodes of text emotions pattern at time t, U_{for} denotes the learning rate of the forget gate, the h_{t-1} represents the currently hidden nodes input value and the bi_{for} denotes bias assigned for each forget node. The forget gate node calculates the sigmoid value for each input text emotional conversation data and currently hidden nodes and parameters values. Finally, the sigmoid gate decides whether to keep node value or ignore based on the state activation threshold. In this research, this research gap in earlier study was resolved by a thorough learning methodology. The deep learning models (DNN) analyze students' textual input so that they can automatically recognise the emotional state of each student and provide their feedback independently to each student group. The profound model includes several phases such as text data pre-processing, text extracting features, a selection of features from the derived emotional feature set and a profound classifier to identify the various emotions of the students. The next section examines the emotional detection models' techniques:

$$\begin{cases} \hat{h}_t = \partial_h(W_hA_t + U_h(for_t * h_{t-1}) + bi_h) \\ \partial(av) = \frac{\sin av}{\cos av} = \frac{e^{av} - e^{-pv}}{e^{av} + e^{-pv}} \end{cases} \quad (11)$$

In eqn(11) the \hat{h}_t denotes the candidate vector value at time t and the symbol ∂_h denotes the tangent activation function gate for each candidate, ac denotes the actual class value of text emotion, and pc denotes the predicted class value. This activation function decides the activation by using $\partial(av)$ value, which is calculated by dividing the $\sin(av)$ value and $\cos(av)$ value. The tangent activation's default threshold range is denoted as tangent $(-1, 1)$, -1 represents an unsuccessful node, and 1 represents a successful node, but the values between >-1 and <1 are considered for back-propagation. The ' $W_h A_t$ ' denotes, the weight of each hidden candidate output node's output of text emotions pattern at time t , and the b_{i_h} denotes bias assigned for each hidden candidate node's output. The candidate vector calculates the tangent value for each text data along with the current hidden node's value and forgets the node's value for each timestamp. Finally, the tangent gate decides whether to keep node value or ignore based on the state activation threshold:

$$o_t = (1 - for_t) * h_{t-1} + for_t * \hat{h}_t \quad (12)$$

In eqn(12) the o_t denotes the output gate value at time t and the symbol \hat{h}_t denotes the candidate vector value at time t , the W_h denotes the weight of each hidden candidate output node's output, and the b_{i_h} denotes bias assigned for each hidden candidate node output. Finally, the output gate node learns emotional text groups' features successful based on forget node values and candidate vector value after the element wise multiplications. The output node stores predicted values for each text conversation data. The predicted value of the sigmoid gate and the tangent gate is near the activation threshold (appropriate text emotional class values); otherwise, it back-propagates the network using FSO until the specific timestamp gets over. The GRU network is a long-term short-term memory (LSTM) network add-on. GRU does similar work and LSTM however the cell layer is not used to transport data. The GRU network has various advantages over LSTM, including lower and quicker efficiency. The current input is learned from its hidden layer node output throughout the network train. The GU is the same as the fully linked GU unit. This network still works somewhat differently from GRU which is fully linked and fuses as forgotten gate to the updating and reset gate vector.

The fish swarm optimizer FSO has been incorporated in GRUNN model to update nodes weight by back-propagating the nodes during the model training. The fish swarm's combined outcomes are target behaviour, random behaviour, teeming behaviour, tracking based convergence, and random behaviour. Position of the input node is represented as a text feature vector (attribute) A :

$$A = (a_{m1}, a_{m2}, \dots, a_{mn}) \quad (13)$$

Eqn(13), input text feature vector A contains the extracted text feature information. The fitness function of the target (B) denoted as A_m :

$$d_{mn} = a_n - a_m \quad (14)$$

Hence, in eqn(14), the derivation of is used to calculate the distance between n and m of a specific attribute.

The target searching behaviour contains two conditions, in first condition if $f(B_n) = f(A_m)$ then, it is considered as convergence. So it takes values towards A_m to A_n , otherwise randomly choose next state A_n :

$$\vec{A}_m = \begin{cases} \text{if } f(B_n) < f(A_m), & A_m + \text{step } a \frac{A_n - A_m}{d_{mn}} \\ \text{else,} & \text{random search} \end{cases} \quad (15)$$

where \vec{A}_m denotes the new state of the input attribute and the random interval value (0,1). The eqn(15) is used to calculate the target behaviour function-based convergence.

This phase takes fitness value for centroid (B_{cen}) of the target attribute or neighbour attribute value Nei_{iatr} and the group factor of the target attribute. Checking $\text{if } f(B_{cen}) / Nei_{iatr} < \partial aB_n$, it takes the centroid (cen) of attributes; else, it remains in the target position. Where ∂ represents group factor, the range values assigned as $\in (0,1)$. The mathematical notation of the teeming behaviour-based optimization is denoted as follows:

$$\vec{A}_m = \begin{cases} \text{if } f(B_{cen}) / Nei_{iatr} < \partial aB_n, & A_m + \text{step } a \frac{A_{cen} - A_m}{d_{m,cen}} \\ \text{else,} & \text{targeting position} \end{cases} \quad (16)$$

where eqn(16) is used to calculate the teeming behaviour-based function to convergence.

The phase computes local minima of a current neighbour of A_m based convergence approach. It checks if the target attributes local minima lm of current neighbours value is $< \partial aB_m$ then takes towards local minima of current neighbour; otherwise, it chooses target behaviour value. The mathematical notation of the tracking behaviour-based optimization is denoted as follows:

$$\vec{A}_m = \begin{cases} \text{if } f(B_{lm}) / Nei_{fish} < \partial aB_m, & A_m + \text{step } a \frac{A_{lm} - A_m}{d_{m,lm}} \\ \text{else,} & \text{targeting position} \end{cases} \quad (17)$$

In eqn(17) is used to calculate the tracking based convergence functions. Generally, the fish selects position by visual range; likewise, the GRUNN classifier's weight of nodes updating position value by choosing any one of the local minima of a neighbour as convergence value. During the model training, whenever the new text emotion arrives for training or testing, the new direction-finding behaviours (in eqn(20)) of FSO in the GRUNN classifier help back-propagate the network node and update the weight to predict the most possible matching text emotion.

The above algorithm explains the step-by-step text analysis process for emotional group detection. Initially, the input text conversations datasets are collected from various data sources. The raw text data has been pre-processed to remove unwanted noise by using multiple NLP pre-processing steps. Next, the text features are extracted from the pre-processed text data using tf-idf extractor method. The extracted text-based feature dimensions are reduced by using chi-square based ranking approach. Finally, the selected features are taken as input to the FSOGRU network model. The GRU computes the sigmoid and tangent value for each information to active candidate vector input. The forget gate helps to remember the previous emotion class value to train the current emotion class. The candidate vector checks the current text match with any of the previously trained class value or not. If matches are found, then the output gate activates appropriate emotion class and sends feedback to all students in an emotional group. Otherwise, back-propagate the network nodes and update the node's weight

Algorithm 1. Deep FSOGRUNN algorithm

Input: text document / conversations

$t = A_{ij}$, $h = 0$, $bi = , U=0.02$ // Initialize the population size, hidden layer, bias value and learning rate

FOR each i=1 to m

FOR each j=1 to n

A=Remove_panch(A) // remove punctuations

A= Lower_case(A) // convert it as lowercase

A=Word_Slicing(A) // word splitting

A=padding(A) // zero padding

ExFeaV= Tf-idf(A_{ij}) // text feature extractions

NormalizedAtributes values (NAV)= min_max(ExFeaV) // normalization

ChiSqr =Chi_square(NAV) // chi squared // Feature selections

Scr= Scoring (ChiSqr) // ranking // Feature selections

Inter_corre = Inter_correlations(Scr) // subset of contents. // Feature selections

$$A_{ij} = Inter_corre$$

END IF

END FOR

END FOR

FOR each t=1 to m

FOR each h=1 to n

$\sigma_g (W_{for} A_t + U_{for} h_{t-1} + bi_{for})$ //Compute the sigmoid value for forget gate value

$\partial_h (W_h A_t + U_h (for_t * h_{t-1}) + bi_h)$ //Compute the tangent value for candidate gate value

IF ($\partial_h == 1$) && ($\sigma_g == 1$)

for_t // forget state activate

\hat{h}_t // candidate vector state activate

$o_t = (1 - for_t) * h_{t-1} + for_t * \hat{h}_t$ // update output date value

ELSE IF ($\partial_h > -1$) && ($\sigma_g > 0$)

FOR each i=1 to m

FOR each j=1 to n

$$\vec{A}_m = \begin{cases} if f(B_{lm}) / Nei_{fish} < \partial aB_m, & A_m + step a \frac{A_{lm} - A_m}{d_{m,lm}} \\ else, & targeting\ position \end{cases}$$

$A_i(t+1) = \vec{A}_m$ //update network nodes weight by using the FSO-based back-propagating

END FOR

END FOR

for_t // forget state activate

\hat{h}_t // candidate vector state activate

$o_t = (1 - for_t) * h_{t-1} + for_t * \hat{h}_t$ // update output value

ELSE

$o_t = 0$ // ignore

END IF

END FOR

END FOR

Output: Detected text emotional(Angry, Sad, Happy, Fear) groups.

using FSO until each input nodes reaches the desired output (emotional class). The subsequent section discusses the efficiency of the deep classification model. Unwanted text information is deleted from the original text documents in text pre-processing. Pre-processing consists of three phases: elimination of punctuation, transformation of the lower case, word slicing. The punctuations and certain characters of your text document will be deleted in the first step; subsequently, the punctuation free phrase will be transformed. Finally, with the aid of space each word in a phrase is cut. Then all the words in a phrase are added to match '0' padding.

4. RESULTS AND DISCUSSIONS

This section discusses the Deep FSOGRUNN model-based text emotion detection approach and the evaluation performance evaluation results. The classifier's efficiency is compared with other deep models based on the text emotion classification approach discussed in section 2. In this research, the Deep FSOGRUNN efficiency is compared with Bi-LSTM(20), BERT-LSTM(21), Bi-GRUN(31). The following evaluation metrics are utilized to evaluate Deep FSOGRUNN performance:

$$\text{Accuracy (Acc)} = \frac{\text{TruePositive} + \text{TrueNegative}}{\text{TruePositive} + \text{TrueNegative} + \text{FalseNegative} + \text{FalsePositive}} \quad (18)$$

$$\text{Precision (Pr)} = \frac{\text{TruePositive}}{\text{TruePositive} + \text{FalsePositive}} \quad (19)$$

$$\text{TruePositive (TP)} = \frac{\text{TruePositive}}{\text{TruePositive} + \text{FalseNegative}} \quad (20)$$

$$\text{FalsePositive (FP)} = \frac{\text{FalsePositive}}{\text{TruePositive} + \text{FalsePositive}} \quad (21)$$

$$\text{Rootmeansquareerror (r}^2\text{)} = \sqrt{\frac{\sum_{r=1}^N (AV_r - PV_r)^2}{N}} \quad (22)$$

$$\text{mc_LogLoss} = -\sum_{\mathcal{L}=1}^{tc} a_o, \mathcal{L} \log(p_o, \mathcal{L}) \quad (23)$$

where in eqn(22) AV_r denotes the actual value of the class label in r^{th} position and PV_r denotes predicted value in r^{th} position. N and n indicate the total number of gene patterns. The eqn(18),eqn(19),

and eqn(20) are various denotes accuracy matrices, which are used to calculate the efficiency values and eqn(21), eqn(22) are denotes indicating error rate metrics, which are used to calculate the error values. The eqn(23) is utilized to calculate the sum of log loss values across classes for each class prediction during the model training. Where ' $\mathcal{L} = 1, 2, \dots, tc$ ' denotes the dimension n of visible nodes. The ' ρ_o ' represents the probability observation of the predicted text emotion class and a_o is correctly predicted emotion class label of the observation 'o'. In the multi-class classification, \mathcal{L} represents the class label, and the ' tc ' denotes the total number of class labels.

Table 3 contains DFSOGRUNN model's accuracy values for six different text emotion datasets. During the DFSOGRUNN model training and testing, the EmoInt (A), EmoBank (B), Preotiuc-Pietro (C), crowded flower(D), ISEAR (E), and SemEval Stance(F) datasets have obtained the maximum of 98.93%, 99.91%, 99.93%, 99.56%, 99.52%, and 99.91% average accuracy values respectively. All text emotion datasets the Preotiuc-Pietro (C) are obtained a maximum of 99.93% accuracy rate.

Figure 3 illustrate the accuracy curves of Deep FSOGRUNN classifier for six datasets. It clearly shows that the combined update and reset gates feature in forget gate of GRUNetwork model $for_t = \sigma_g \left(W_{for} A_t + U_{for} h_{t-1} + bi_{for} \right)$ helps to remember the emotion class values efficiently for all the text emotion dataset.

Figure 4 demonstrates the accuracy value obtained by BERT-BILST, CNN-BiGRUN, CNN-BiLSTM, FSOGRUNN classifiers. It clearly shows that the proper utilization of Tf-idf based feature extraction $\left(M(T; DP) = \frac{1}{|D|} \sum_{t,d} tf(t, d) * \frac{1}{|D|} * idf(t) \right)$ in FSOGRUN classifier has been obtained higher accuracy rate (99.52%) in identifying emotional groups than comparison approaches.

Table 3. Summary of text emotion detection accuracy for Deep FSOGRUNN model (%)

Batch Size	A	B	C	D	E	F
25	98.61	99.14	99.52	99.23	99.52	99.55
50	98.93	99.71	99.33	99.16	99.34	99.22
75	98.74	99.91	99.46	98.88	99.45	99.91
100	98.32	99.22	99.93	98.93	99.29	99.72
150	98.34	99.51	99.85	99.56	99.39	99.79

Note*: The value in bold letter denotes the maximum accuracy values

Figure 3. Accuracy value of Deep-FSOGRUNN classifier for six dataset

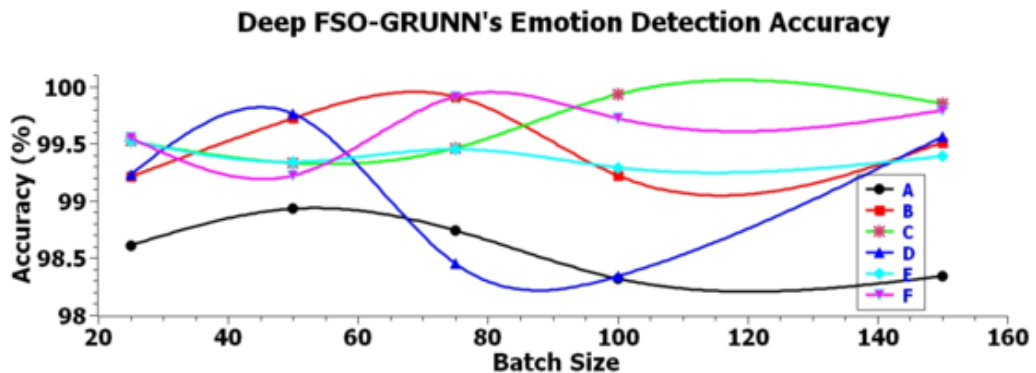


Figure 4. Accuracy comparison curve for ISEAR dataset

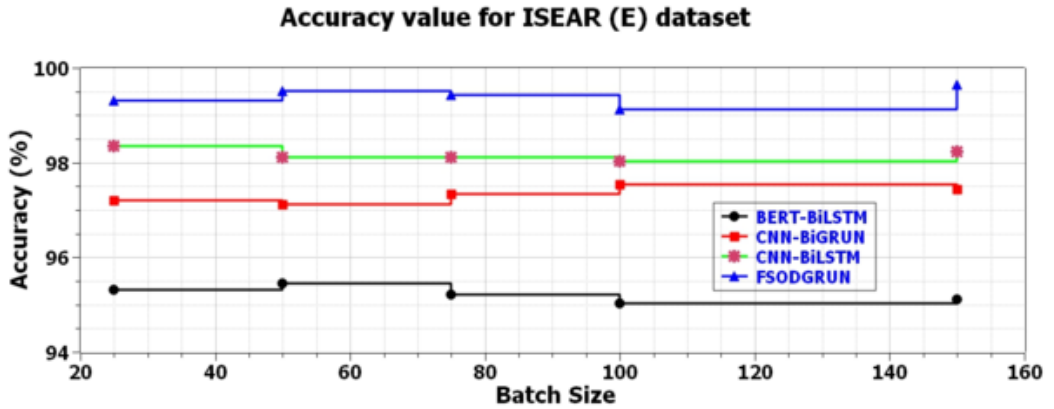


Figure 5 demonstrates the F-1 score value obtained by BERT-BiLST, CNN-BiGRUN, CNN-BiLSTM, FSODGRUN classifiers. It clearly shows that the proper utilization of NLP based pre-processing steps in FSODGRUN classifier has been obtained higher f1-score rate (99.39%) in identifying emotional groups than comparison approaches. It helps to allocate feedback to appropriate emotional group students.

Figure 6 demonstrates the precision value obtained by BERT-BiLST, CNN-BiGRUN, CNN-BiLSTM, FSODGRUN classifiers. The FSODGRUN classifier is obtained higher precision rate (99.44%) in identifying emotional groups than comparison approaches. It clearly shows that the proper utilization Chi-square based feature ranking approach $\left(|sv_a| = \sqrt{(igs_a)^2 + (chis_a)^2} \right)$ fit with FSODGRUN classifier to select the significant feature from the extracted text emotions details.

Figure 7 demonstrates the Root mean square error value obtained by BERT-BiLST, CNN-BiGRUN, CNN-BiLSTM, FSODGRUN classifiers. The FSODGRUN classifier has been obtained less error rate RMSE (0.44%) in identifying emotional groups than comparison approaches. It clearly shows that the tracking behaviour-based fitness function of FSO:

Figure 5. F-1 score comparison curve for EmoInt dataset

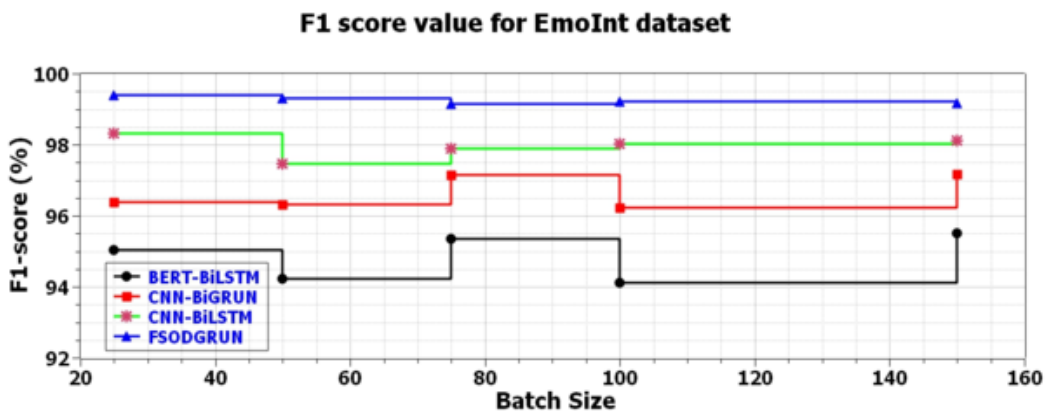


Figure 6. Precision rate comparison curve for EmoBank dataset

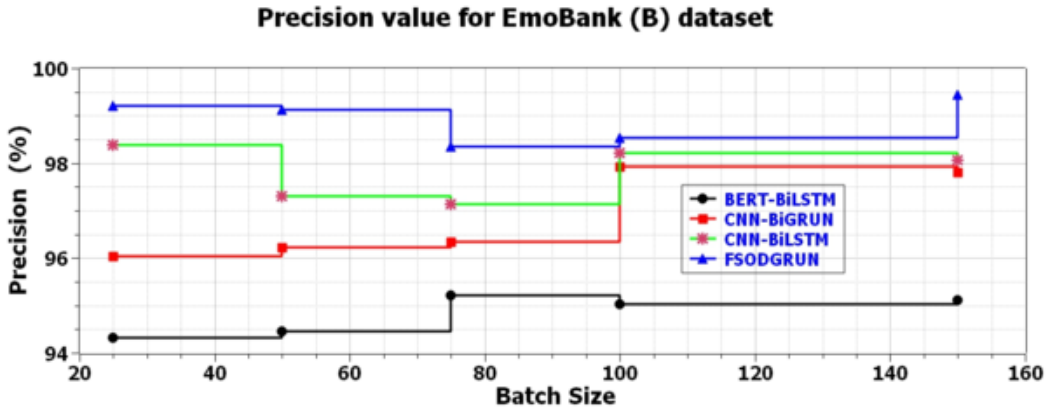
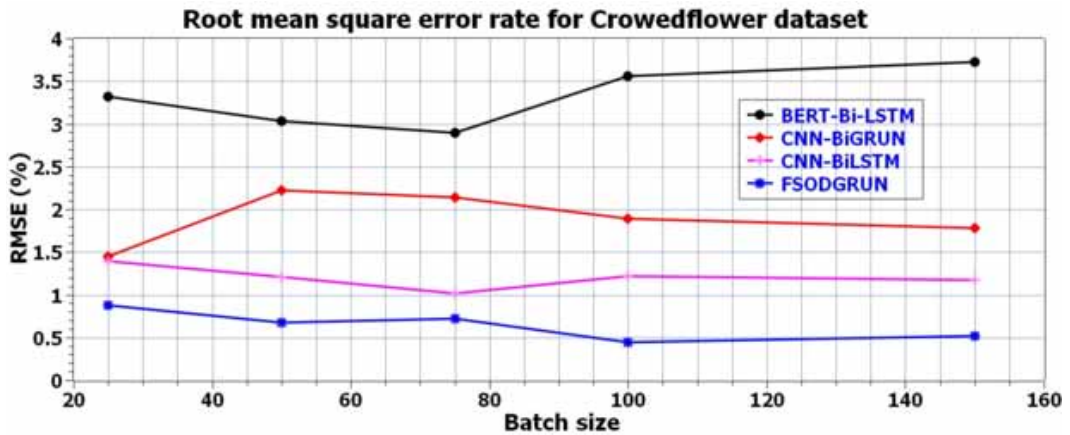


Figure 7. Root mean square error comparison curve for Crowed flower dataset



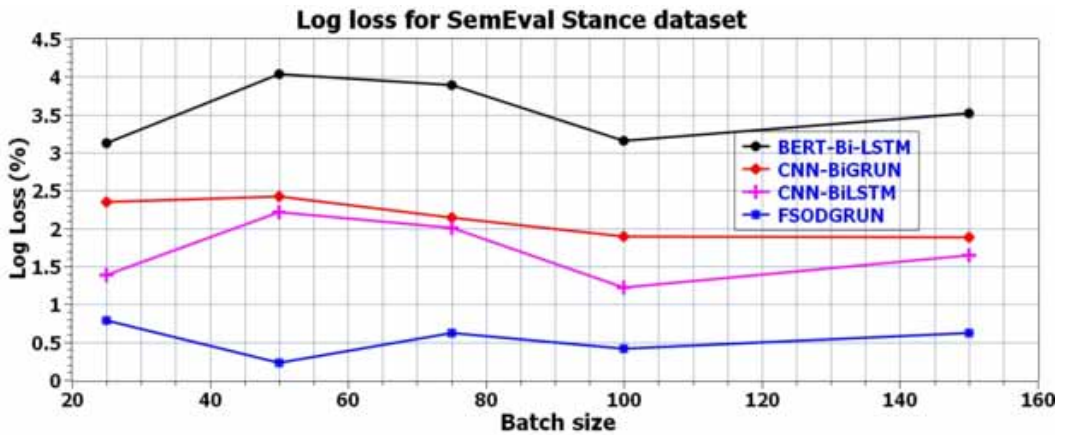
$$\vec{A}_m = \begin{cases} \text{if } f(B_{lm}) / Nei_{fish} < \partial aB_m, & A_m + \text{step } a \frac{A_{lm} - A_m}{d_{m,lm}} \\ \text{else,} & \text{targeting position} \end{cases}$$

in GRUN updated the network weight correctly during the model training.

Figure 8 demonstrates the Log loss value obtained by BERT-BiLST, CNN-BiGRUN, CNN-BiLSTM, FSODGRUN classifiers. The FSODGRUN classifier has been obtained less loss rate (0.23%) by using the $mc_LogLoss = -\sum_{\mathcal{L}=1}^{tc} a_o, \mathcal{L} \log(p_o, \mathcal{L})$ derivation. It clearly shows that the proper pre-processing and feature extraction techniques in FSODGRUN help identify emotional groups instead of comparison approaches.

This section's overall results and discussions prove that the FSODGRUN model has been fulfilled the research gaps in earlier studies, like covering emotional relationships between words. The

Figure 8. Log Loss rate comparison curve for SemEval dataset



maximum accuracy (99.93%) and reduced loss rate (0.23%) proves that the classifier achieved the main objectives, such as improving efficiency and loss rate in emotion detection.

5. CONCLUSION

Thus, the above section discusses the evaluation results and efficiency of the FSOGRUN classifier. The proper utilization of NLP steps in the deep model helps cover the emotional relationship between words during the FSOGRUN model training and testing. The EmoInt (A), EmoBank (B), Preotiuc-Pietro (C), crowded flower(D), ISEAR (E), and SemEval Stance(F) datasets are obtained the maximum of 98.93%, 99.91%, 99.93%, 99.56%, 99.52%, and 99.91% average accuracy values respectively. Moreover, the tracking behaviour based fitness function of FSO in GRU-network helps obtain a minimum of 0.23% loss rate and a maximum of 99.93% accuracy than other classifiers. It proves that deep FSOGRU network classifier outperforms in detecting various emotional group students from text conversations to allocate feedback messages. The FSOGRUN system is designed to monitor students' emotions by retrieving text responses from the Chabot and sending feedback messages to each emotion group. This text emotion detection helps to improve the learning process and reduces the difficulties in e-learning as well. The research has been extended to detect other human behaviour like opinion and sentiment analysis in the future.

REFERENCES

- Abdullah, M., Hadzikadicy, M., & Shaikhz, S. (2018). SEDAT: Sentiment and Emotion Detection in Arabic Text Using CNN-LSTM Deep Learning. *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 835-840. doi:10.1109/ICMLA.2018.00134
- Ahmad, U., Song, H., Bilal, A., Mahmood, S., & Alazab, M. (2021). A Novel Deep Learning Model to Secure Internet of Things in Healthcare. *Machine Intelligence and Big Data Analytics for Cybersecurity Applications*, 341-353.
- Al-Omari, H., Abdullah, M. A., & Shaikh, S. (2020). EmoDet2: Emotion Detection in English Textual Dialogue using BERT and BiLSTM Models. *2020 11th International Conference on Information and Communication Systems (ICICS)*, 226-232. doi:10.1109/ICICS49469.2020.239539
- Al-Turjman, F., Zahmatkesh, H., & Mostarda, L. (2019). Quantifying uncertainty in internet of medical things and big-data services using intelligence and deep learning. *IEEE Access: Practical Innovations, Open Solutions*, 7, 115749–115759. doi:10.1109/ACCESS.2019.2931637
- Alarifi, A., & Alwadain, A. (2020). Killer heuristic optimized convolution neural network-based fall detection with wearable IoT sensor devices. *Measurement*, 108258. Advance online publication. doi:10.1016/j.measurement.2020.108258
- Alazab, Jolfaei, Vinayakumar, Soman, & Poornachandran. (2019). DBD: Deep learning DGA-based botnet detection. *Deep Learning Applications for Cyber Security*.
- Batbaatar, E., Li, M., & Ryu, K. H. (2019). Semantic-Emotion Neural Network for Emotion Recognition from Text. *IEEE Access: Practical Innovations, Open Solutions*, 7, 1–1. doi:10.1109/ACCESS.2019.2934529
- Buechel & Hahn. (2017). EmoBank: Studying the Impact of Annotation Perspective and Representation Format on Dimensional Emotion Analysis. In *EACL 2017 - Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics. Valencia, Spain, April 3-7, 2017. Volume 2, Short Papers* (pp. 578-585). Available: <http://aclweb.org/anthology/E17-2092>
- Buechel & Hahn. (2017). Readers vs writers vs. texts: Coping with different perspectives of text understanding in emotion annotation. In *LAW 2017 - Proceedings of the 11th Linguistic Annotation Workshop @ EACL 2017*. Available: <https://sigann.github.io/LAW-XI-2017/papers/LAW01.pdf>
- Chatterjee, A., Gupta, U., Chinnakotla, M. K., Srikanth, R., Galley, M., & Agrawal, P. (2019). Understanding Emotions in Text Using Deep Learning and Big Data. *Computers in Human Behavior*, 93, 309–317. doi:10.1016/j.chb.2018.12.029
- Chen, L. P., Lin, J. A., Li, K. C., Hsu, C. H., & Chen, Z. X. (2014). A scalable blackbox-oriented e-learning system based on desktop grid over private cloud. *Future Generation Computer Systems*, 38, 1–10. doi:10.1016/j.future.2014.02.017
- Cheng, Y., Yao, L., Xiang, G., Zhang, G., Tang, T., & Zhong, L. (2020). Text Sentiment Orientation Analysis Based on Multi-Channel CNN and Bidirectional GRU With Attention Mechanism. *IEEE Access: Practical Innovations, Open Solutions*, 8, 134964–134975. doi:10.1109/ACCESS.2020.3005823
- Clarizia, F., Colace, F., De Santo, M., Lombardi, M., Pascale, F., & Pietrosanto, A. (2018). E-learning and sentiment analysis. *Proceedings of the 6th International Conference on Information and Education Technology - ICJET '18*. doi:10.1145/3178158.3178181
- Dzisevič, R., & Šešok, D. (2019). Text Classification using Different Feature Extraction Approaches. *2019 Open Conference of Electrical, Electronic and Information Sciences (eStream)*, 1-4. doi:10.1109/eStream.2019.8732167
- El Hammoumi, O., Benmarrakchi, F., Ouherrou, N., El Kafi, J., & El Hore, A. (2018). Emotion Recognition in E-learning Systems. *2018 6th International Conference on Multimedia Computing and Systems (ICMCS)*, 1-6. doi:10.1109/ICMCS.2018.8525872
- El Hammoumi, O., Benmarrakchi, F., Ouherrou, N., El Kafi, J., & El Hore, A. (2018). Emotion Recognition in E-learning Systems. *2018 6th International Conference on Multimedia Computing and Systems (ICMCS)*, 1-6. doi:10.1109/ICMCS.2018.8525872

- Elmasry, W., Akbulut, A., & Zaim, A. H. (2018). Deep learning approaches for predictive masquerade detection. *Security and Communication Networks*, 2018, 1–24. doi:10.1155/2018/9327215
- Ghanbari, A. M., Ghanbari, S., & Norouzi, Y. (2017). A new approach to architecture of human-computer interaction. *2017 IEEE 4th International Conference on Smart Instrumentation, Measurement and Application (ICSIMA)*, 1-4. doi:10.1109/ICSIMA.2017.8311991
- Hakdağlı, Ö., Özcan, C., & Oğul, İ. Ü. (2018). Stream text data analysis on twitter using apache spark streaming. *2018 26th Signal Processing and Communications Applications Conference (SIU)*, 1-4. doi:10.1109/SIU.2018.8404540
- Haq, A. U., & Li, J. (2020). *A New Intelligent Approach for Effective Recognition of Diabetes in the IoT E-HealthCare Environment*. 10.20944/preprints202002.0462.v
- Hulliyah, K., Awang Abu Bakar, N. S., & Ismail, A. R. (2017). Emotion recognition and brain mapping for sentiment analysis: A review. *2017 Second International Conference on Informatics and Computing (ICIC)*, 1-5. doi:10.1109/IAC.2017.8280568
- Iqbal, A., & Barua, K. (2019). A Real-time Emotion Recognition from Speech using Gradient Boosting. *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, 1-5. doi:10.1109/ECCE.2019.8679271
- Jacob, S., Menon, V. G., Al-Turjman, F., & Vinoj, P. G. (2019). Artificial muscle intelligence system with deep learning for post-stroke assistance and rehabilitation. *IEEE Access: Practical Innovations, Open Solutions*, 7, 133463–133473. doi:10.1109/ACCESS.2019.2941491
- Jafar, S., & Shaikh, T. (2019). Image Based Emotion Detection. *2019 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE)*, 323-328. doi:10.1109/ICCIKE47802.2019.9004259
- Kamalov, F., & Thabtah, F. (2017). A Feature Selection Method Based on Ranked Vector Scores of Features for Classification. *Annals of Data Science*, 4(4), 483–502. doi:10.1007/s40745-017-0116-1
- Karna, M., Juliet, D. S., & Joy, R. C. (2020). Deep learning based Text Emotion Recognition for Chatbot applications. *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184)*, 988-993. doi:10.1109/ICOEI48184.2020.9142879
- KP, S., & Alazab, M. (2020). *A Comprehensive Tutorial and Survey of Applications of Deep Learning for Cyber Security*. Academic Press.
- Kratzwald, B., Ilić, S., Kraus, M., Feuerriegel, S., & Prendinger, H. (2018). Deep learning for affective computing: Text-based emotion recognition in decision support. *Decision Support Systems*, 115, 24–35. Advance online publication. doi:10.1016/j.dss.2018.09.002
- Kwon, J., Kim, D., Park, W., & Kim, L. (2016). A wearable device for emotional recognition using facial expression and physiological response. *38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 5765-5768. doi:10.1109/EMBC.2016.7592037
- Majumder, N., Poria, S., Gelbukh, A., & Cambria, E. (2017). Deep Learning-Based Document Modeling for Personality Detection from Text. *IEEE Intelligent Systems*, 32(2), 74–79. doi:10.1109/MIS.2017.23
- Manogaran, G., Shakeel, P. M., Fouad, H., Nam, Y., Baskar, S., Chilamkurti, N., & Sundarasekar, R. (2019). Wearable IoT smart-log patch: An edge computing-based Bayesian deep learning network system for multi access physical monitoring system. *Sensors (Basel)*, 19(13), 2019. doi:10.3390/s19133030 PMID:31324070
- Mohaouchane, H., Mourhir, A., & Nikolov, N. S. (2019). Detecting Offensive Language on Arabic Social Media Using Deep Learning. *2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS)*, 466-471. doi:10.1109/SNAMS.2019.8931839
- Mukhopadhyay, S., & Sharma, S. (2020). Real-Time Facial Expression and Emotion Recognition using Eigenfaces, LBPH and Fisher Algorithms. *2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, 212-220. doi:10.1109/Confluence47617.2020.9057985
- Muthu, BA., CB, S., Kumar, P.M., Kadry, S.N., & Hsu, C.H. (n.d.). A Framework for Extractive Text Summarization based on Deep Learning Modified Neural Network Classifier. *ACM Transactions on Asian and Low-Resource Language Information Processing*.

- Nasim, Z., Rajput, Q., & Haider, S. (2017). Sentiment analysis of student feedback using machine learning and lexicon based approaches. *2017 International Conference on Research and Innovation in Information Systems (ICRIIS)*, 1-6. doi:10.1109/ICRIIS.2017.8002475
- Naz, H., & Ahuja, S. (2020). Deep learning approach for diabetes prediction using PIMA Indian dataset. *Journal of Diabetes and Metabolic Disorders*, 19(1), 391–403. Advance online publication. doi:10.1007/s40200-020-00520-5 PMID:32550190
- Park, C.-H., & Sim, K.-B. (2003). Emotion recognition and acoustic analysis from speech signal. *Proceedings of the International Joint Conference on Neural Networks*, 2594–2598. doi:10.1109/IJCNN.2003.1223975
- Rajalingam, B., Al-Turjman, F., & Santhoshkumar, R. (2020). Intelligent multimodal medical image fusion with deep guided filtering. *Multimedia Systems*, 1–15.
- Sailunaz, K., & Alhadj, R. (2019). Emotion and Sentiment Analysis from Twitter Text. *Journal of Computational Science*, 36, 101003. Advance online publication. doi:10.1016/j.jocs.2019.05.009
- Serte, S., Serener, A., & Al-Turjman, F. (2020). Deep learning in medical imaging: A brief review. *Transactions on Emerging Telecommunications Technologies*, 4080.
- Shatnawi, Abdullah, & Hammad. (2020). MLEngineer at SemEval-2020 Task 7: BERT-Flair Based Humor Detection Model. *International Committee for Computational Linguistics*, 1041-1048.
- Sueno, Gerardo, & Medina. (2020). Multi-class Document Classification using Support Vector Machine (SVM) Based on Improved Naïve Bayes Vectorization Technique Methodology. *IJATCSE*, 9(3).
- Tai, D.W.S., Zhang, R.C., Chang, S.H., Chen, C.P., & Chen, J.L. (2012). A meta-analytic path analysis of e-learning acceptance model. *International Journal of Educational and Pedagogical Sciences*.
- Tandel, S. S., Jamadar, A., & Dudugu, S. (2019). A Survey on Text Mining Techniques. *2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS)*, 1022-1026. doi:10.1109/ICACCS.2019.8728547
- Ullah, F., Jabbar, S., & Al-Turjman, F. (2020, October). Programmers' de-anonymization using a hybrid approach of abstract syntax tree and deep learning. *Technological Forecasting and Social Change*, 159, 120186. doi:10.1016/j.techfore.2020.120186
- Ullah, F., Naeem, H., Jabbar, S., Khalid, S., Latif, M. A., Al-turjman, F., & Mostarda, L. (2019). Cyber security threats detection in internet of things using deep learning approach. *IEEE Access: Practical Innovations, Open Solutions*, 7, 124379–124389. doi:10.1109/ACCESS.2019.2937347
- Ullah, F., Wang, J., Jabbar, S., Al-Turjman, F., & Alazab, M. (2019). Source Code Authorship Attribution Using Hybrid Approach of Program Dependence Graph and Deep Learning Model. *IEEE Access: Practical Innovations, Open Solutions*, 7, 141987–141999. doi:10.1109/ACCESS.2019.2943639
- Yadollahi, A., Shahraki, A. G., & Zaiane, O. R. (2017). Current State of Text Sentiment Analysis from Opinion to Emotion Mining. *ACM Computing Surveys*, 50(2), 1–33. doi:10.1145/3057270
- Ye, F., Shao, S., & Tian, Y. (2018). Weapon Target Assignment Based on Improved Artificial Fish Swarm Algorithm. *2018 USNC-URSI Radio Science Meeting (Joint with AP-S Symposium)*, 15-16. doi:10.1109/USNC-URSI.2018.8602636
- Yi, S., & Liu, X. (2020). Machine learning-based customer sentiment analysis for recommending shoppers, shops based on customers' review. *Complex Intell. Syst.*, 6(3), 621–634. doi:10.1007/s40747-020-00155-2
- Yoon, S., Byun, S., Dey, S., & Jung, K. (2019). Speech Emotion Recognition Using Multi-hop Attention Mechanism. *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2822-2826. doi:10.1109/ICASSP.2019.8683483
- Young, T., Hazarika, D., Poria, S., & Cambria, E. (2018, August). Recent Trends in Deep Learning Based Natural Language Processing (Review Article). *IEEE Computational Intelligence Magazine*, 13(3), 55–75. doi:10.1109/MCI.2018.2840738
- Zobeidi, S., Naderan, M., & Alavi, S. E. (2017). Effective text classification using multi-level fuzzy neural network. *2017 5th Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS)*, 91-96. doi:10.1109/CFIS.2017.8003664

S. Kirubakaran is working as Professor in the Department of Computer Science and Engineering in Jayamukhi Institute of Technological Sciences, Telangana State, India. He is having 15 years of teaching experience and 10 years of research experience. He has completed his PhD under Faculty of Information and Communication Engineering in Anna University Chennai, Tamilnadu, India in the year of 2013. He has completed his Master of Engineering (CSE) in Muthayammal Engineering College, Tamilnadu, India in the year of 2008. He has completed his Bachelor of Engineering (CSE) in Mahendra Engineering College, Tamilnadu, India in the year of 2004. He is a recognized supervisor in Anna University Chennai. He is peer reviewer in many refereed journals. He has published 24 International journals and published book which titled as "Mobile Computing". He published two Indian patents. He has presented 12 International conferences. He is having professional body membership in ISTE (Life Time), IAENG, IACSIT.