Computers, Materials & Continua
DOI:10.32604/cmc.2022.021671
Article

Tech Science Press

BERT-CNN: A Deep Learning Model for Detecting Emotions from Text

Ahmed R. Abas[1], Ibrahim Elhenawy[1], Mahinda Zidan[2,*] and Mahmoud Othman[2]

[1]Department of Computer Science, Faculty of Computer and Informatics, Zagazig University, Zagazig, 44519, Egypt
[2]Department of Computer Science, Faculty of Computers and Information Technology, Future University in Egypt, 11835, Egypt
*Corresponding Author: Mahinda Zidan. Email: mahinda.zidan@fue.edu.eg
Received: 10 July 2021; Accepted: 27 September 2021

Abstract: Due to the widespread usage of social media in our recent daily lifestyles, sentiment analysis becomes an important field in pattern recognition and Natural Language Processing (NLP). In this field, users' feedback data on a specific issue are evaluated and analyzed. Detecting emotions within the text is therefore considered one of the important challenges of the current NLP research. Emotions have been widely studied in psychology and behavioral science as they are an integral part of the human nature. Emotions describe a state of mind of distinct behaviors, feelings, thoughts and experiences. The main objective of this paper is to propose a new model named BERT-CNN to detect emotions from text. This model is formed by a combination of the Bidirectional Encoder Representations from Transformer (BERT) and the Convolutional Neural networks (CNN) for textual classification. This model embraces the BERT to train the word semantic representation language model. According to the word context, the semantic vector is dynamically generated and then placed into the CNN to predict the output. Results of a comparative study proved that the BERT-CNN model overcomes the state-of-art baseline performance produced by different models in the literature using the semeval 2019 task3 dataset and ISEAR datasets. The BERT-CNN model achieves an accuracy of 94.7% and an F1-score of 94% for semeval2019 task3 dataset and an accuracy of 75.8% and an F1-score of 76% for ISEAR dataset.

Keywords: BERT-CNN; deep learning; emotion detection; semeval2019; text classification

1 Introduction

Sentiment analysis is one of the most important tasks of natural language processing (NLP). It can be defined as a process that categorizes and analyzes opinions, sentiments, and emotions towards a company such as organizations, issues, topics, and attributes [1]. Sentiment analysis can determine the polarity of the text whether it is positive, negative, or even neutral by analyzing every word or phrase. It is closely linked to emotion detection. Due to our major, categorization the text into states

of emotion is known as sentiment analysis or emotion detection. The emotional state of the person can be reflected by the text [2]. Various types of emotions such as sadness, fear, anger, and happiness can be found in daily life. Several organizations related to business, psychology, health care, politics, security, and other fields have long tried to extract people's emotions from their social interactions [2]. Because of the need to detect correct emotion in different socio-economic areas, we have developed an automatic emotion detection system. Recently, the use of deep learning approaches that can increase the accuracy of text classification has been growing.

As deep learning advances, neural network architectures e.g., recurrent neural networks (RNN), Convolutional Neural Networks (CNN) [3] and Long Short-Term Memory (LSTM) [4] have demonstrated great performance in solving different natural language processing (NLP) tasks such as language modeling, machine translation, and text classification. In contrast to the success of deep learning in Computer Vision, the success of deep learning models in NLP pales. The lack of large labeled text datasets is one of the major explanations for this slow progress. As the networks have a large number of parameters, overfitting can be caused by training these networks on small datasets. Another big reason why NLP lags behind computer vision is the lack of transfer learning in NLP. Transfer learning [5,6] has played a significant role in the success of computer vision than NLP.

In 2018, Google [7,8] introduced a transformer model, which helps to overcome the transfer learning problem in NLP. Typically, transfer learning is demonstrated by using pre-trained models. A pre-trained model can be defined as a model trained on a big benchmark dataset for solving a problem, which is identical to the one we want to solve. There are several pre-trained language models such as ELMO, GPT [9], and BERT [10]. BERT has accomplished a great achievement on various NLP problems such as Question Answering (QA), Named Entity Recognition (NER), and sentiment analysis. BERT and ELMO rely on the Bidirectional transformer architecture while GPT relies on the left to right transformer architecture.

The Bidirectional Encoder Representations from Transformers [10] known as BERT is used to be pre-trained on English data. It is a group of transformer encoder layers with several heads, for example, fully connected neural networks enhanced by a self-attention mechanism. For every token of input in a series, each head calculates the key-value and query vectors used to produce a weighted representation. The outputs of each head in the same layer are combined and run through a completely linked layer. Each layer is wrapped with a skip connection and a layer normalization is applied after it. Convolutional Neural networks (CNN) are used for text classification [11].

In this paper, we seek to create a model that can produce an emotion from a sentence. We propose a deep active learning model BERT-CNN for emotion detection. This method combines the knowledge embedded in pre-trained deep bidirectional transformer (BERT) with the Convolutional Neural Network (CNN). Based on Semeval 2019, extensive experiments are conducted to be used for emotion detection. State-of - art performance is obtained by our model.

The major contributions of this research work are as follows:

At the beginning of the study, an extensive study and analysis to most recent relevance of Detecting Emotions from Text is introduced.

Second, our work complements a large body of fundamental analysis research that identifies a set of variables that improve assessment [12–33]. Finally, our approach differs from similar studies, where we:

- Proposing the BERT-CNN model that can mark and score any piece of text, especially tweets and social media posts according to three categories of emotions: happiness, anger, and sadness.

To detect these emotions, textual features are extracted; a variety of NLP tools and deep learning classifiers are used.
- Combining the knowledge embedded in the pre-trained deep bidirectional transformer (BERT) with the Convolutional Neural Network (CNN) in the proposed BERT-CNN to detect emotion from the text.

Finally, our approach is distinct from similar studies, according to the word context, the semantic vector is dynamically generated and then placed into the CNN to predict the output,

- Achieving an accuracy of 94.7% and F-measure of 94% by the proposed BERT-CNN Model using the Semeval 2019 task 3 dataset and an accuracy of 75.8% and F-measure of 76% for ISEAR dataset.

The rest of this paper is organized into four main parts: Section 2 introduces the related literature in the field of text classification. Section 3 presents the detailed structure of the proposed model. Section 4 provides the experimental results shown to evaluate and compare the proposed model with other recent deep learning studies. Finally, Section 5 also presents the conclusion and future work.

2 Related Work

This section reviews recent advances in emotion recognition detection and analysis. The previous efforts and the existing studies can be roughly categorized into two categories: First, emotion recognition based on the traditional methods; second, emotion recognition based on deep learning approaches.

2.1 Emotion Recognition Based on Traditional Methods

Asghar et al. [34] classify emotions based on the ISEAR dataset by applying different machine learning algorithms (e.g., SVM, Random Forest, XGboost, KNN, Logistic regression, SGD classifier, and Naive Bayesian) and choose Machine learning algorithm that achieved the high-performance result for emotion detection. The result reveals that, in contrast to other different classifiers, logistic regression had the best result. Limited emotions and authors used only one dataset for experiments, which constitutes the main disadvantage of this research.

Suhasini et al. [35] detected the emotions from twitter messages using supervised machine learning Algorithms. A comparative analysis of two algorithms, Naive Bayes (NB) and K-nearest neighbor (KNN), was performed. The Naïve Bayes (NB) achieved the highest accuracy.

Bhagat et al. [36] classify tweets into three classes, which are positive, negative, and neutral by using a hybrid approach of Naïve Bayes (NB) and K-Nearest Neighbor (KNN). They achieved a good accuracy than other approach like random forest.

Gaind et al. [37] introduce two approaches to classify Twitter texts into six different emotion categories: fear, sadness, happiness, disgust, anger, and surprise. The first approach used natural language processing while the second approach used two machine learning classifiers, which are support vector machine and decision tree classifier. Support vector machine classifier has shown the best results.

2.2 Emotion Recognition Based on Deep Learning

Haryadi et al. [1] use long short-term memory (LSTM) and Nested LSTM to classify text into seven emotions. These emotions are joy, love, fear, thankfulness, anger, sadness, and surprise. Also,

they have compared their results with SVM (Support Vector Machine). The results showed that the best accuracy is achieved using Nested LSTM.

Munikar et al. [4] use the pre-trained BERT model and fine-tune it to solve fine-grained sentiment analysis classification by using SST dataset. The stated techniques were compared with the most popular models in deep learning like RNN, CNN, and LSTM; the comparison showed that their techniques have achieved the best accuracy.

Huang et al. [5] present DCNN-BiGRU (Deep Convolutional Neural Network Bidirectional Gated Recurrent), and it is a model for text classification. This model has based on BERT (Bidirectional Encoder Representations from Transformer) embedding. CCERT email and movie comment datasets are used. The Experiments resulted in an accuracy of 92.66% on the CCERT while 91.89% on the movie comment dataset.

Meng et al. [7] propose a CNN–BILSTM model based on enhanced feature attention. This model is used for aspect-level sentiment analysis. Three datasets are used for model evaluation. The most advanced results are achieved by CNN-BILSTM.

Adoma et al. [12] presented a deep learning model for detecting emotions from text. Their model consisted of two stages: BERT fine-tuned training stage to learn the context of a word considering other words and LSTM classification stage. They classified text into 7 emotions based on ISEAR dataset. They obtained an F-score of 0.73%.

Adoma et al. [13] compared the performance of four pretrained models in bid to determine the best pretrained model for identifying emotions using ISEAR dataset. They compared the performace of of BERT, RoBERT, DistilBERT and xlnet using the same hyper parameters. The results showed a better classification performance in order RoBERT, Xlnet, BERT and DistilBERT. Their results showed an accuracy of e 0.7431, 0.7299, 0.7009, 0.6693 for RoBERTa, XLNet, BERT, and DistilBERT.

Al-Omari et al. [14] proposed a deep learning model to detect four emotions which are happy, sad, angry and other from English text. They applied this model on the semeval task 3 dataset. They extract features using a combination of Glove word embedding, BERT word embedding and set of psycholinguistic features. The proposed system (EmoDet2) is combining a fully connected neural network architecture and BiLSTM neural network to obtain predictions. They obtained an F-Score of 0.748.

Rani et al. [38] introduce a model that combined Glove with parts of speech tagging, aspect embedding, gated recurrent unit, attention mechanism, and convolutional neural network (GPTA-GRUAMCNN). The accuracy was 98.29% on the restaurant reviews dataset.

Ragheb et al. [39] introduce a model formed of two phases to detect and classify emotions from texts. SemEval-2019 Task 3 is the used dataset. This dataset is formed of different conversations that contain different emotions as sadness, anger, happiness, and many other emotions. The results achieved by this model are 75.82%.

Yu et al. [40] BERT is first used to generate word and sentence embeddings for all utterances. The resulting calculated word embeddings are fed into a Convolutional Neural Network (CNN) and its output is then concatenated with the BERT-generated sentence embeddings. Then, the concatenated vectors are used to train a bi-directional GRU with a residual connection followed by a fully connected layer, and finally a SoftMax layer generates predictions that fix class imbalances by using focal loss.

Chiorrini et al. [41] proposed two BERT-based approach for text classification and fine-tuned them which are uncased BERT-base and cased BERT-base in order to evaluate their performance.

They collected data from microblogging platforms and in particular from twitter. They conducted experiments using two different datasets these data sets were used for sentiment analysis and emotion detections. Experiments shows that the proposed models achieved an accuracy of 0.92% for sentiment analysis and 0.90% for emotion recognition. They highlighted that BERT achieves good results in text classification.

Sindhu et al. [42] employ two LSTMs models, which are used for aspect extraction and sentiment detection. Different word embeddings are used to evaluate the model performance. Using the domain embedding as an embedding layer achieved better results. In aspect extraction, the accuracy was 91% while in sentiment detection the accuracy was 93%.

Polignano et al. [43] design a model based on the combination of BiLSTM, CNN, and self-attention. Therefore, the three-word embeddings e.g., Google Word Embedding, Glove Embedding, and Fast Text output were compared. The designed model for evaluation is based on the ISEAR, SemEval2018 Task1, and SemEval-2019 Task 3 datasets.

González et al. [44,45] describe an approach that developed for Contextual emotion detection by establishing an ensemble of the snapshot of 1D Hierarchal CNN to extract features from Semeval 2019 Task3 dataset, propose a model based on the use of a genetic algorithm to ensemble various snapshots of the same model. Also, they introduce a deep learning model for irony detection issues in twitter for English and Spanish languages. The presented work is based on the transformer encoder model to contextualize pre-trained Twitter word embeddings via multi-head scaled dot-product attention mechanisms.

Fei et al. [46] for implicit emotion detection, propose an implicit objective network. During the reconstruction of the input sentence, the vibrational module in this model captures the implicit sentiment target. The classification module leverages such prior knowledge and uses a process of multi-head attention for capturing the clues of the final prediction effectively.

3 Proposed BERT-CNN Deep Learning Model

In this section, the proposed model, BERT-CNN deep learning model, has been clarified (Fig. 1 for systematic visualization). The BERT-CNN primarily consists of three main components: 1) Preprocessing the data. 2) BERT base model, in which the text was passed through 12 layers of self-attention to obtain the contextual vector representation. 3) CNN, which is used as a classifier.

Figure 1: Architecture of the BERT-CNN model

3.1 Pre-processing Data

Preprocessing data is the first step in text classification. It includes the process of cleaning the text by removing the noise data and the uninformative sections of the original text such as hashtags. Preprocessing data increases the efficiency of the classification process. It also includes tokenization, which has the following processes:

- Normalizing URLs, emails, money, date, time, percentage, expressions, and phone numbers.
- Annotating all capitalized letters censored phrases and words with emphasis.
- Annotating and reducing elongated and repeated words

- Unpacking hashtags and contractions.

3.2 Bidirectional Encoder Representations Form Transformers (BERT)

Described as a language model at the lowest layer in the natural language processing (NLP). Through meaningful corpus pre-training, BERT can achieve the greatest global and local feature representations of a sequence. Fig. 2 represent the network structure of the BERT [10]. Which is considered that the network structures of BERT are based on the transformer architecture. Suppose that the dimension of the embedding vector is the input sequence encloses n tokens, the input layer in the BERT model is a matrix, and its output is similarly a matrix. therefore, N BERT layers can be simply connected in series. Where the base model of BERT uses a Transformer block with $N = 12$ layers.

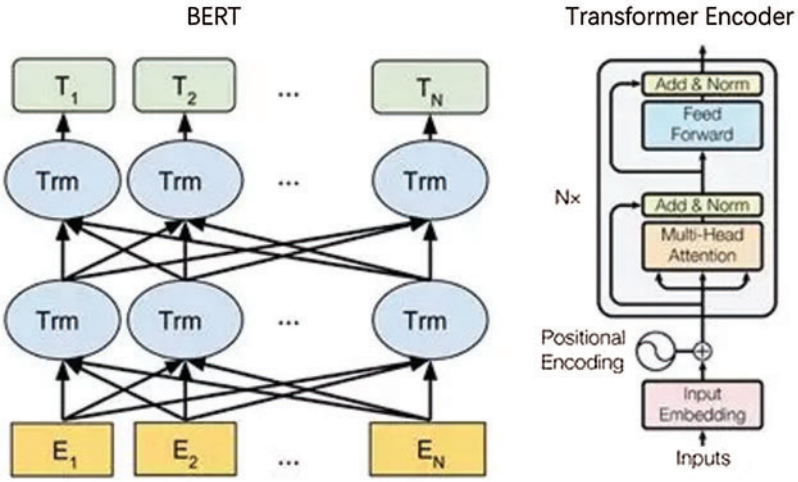

Figure 2: Architecture of the BERT model

3.2.1 Input Representation on BERT

In BERT, the embedding layers are named: Token embeddings, Segment embeddings, and Position embeddings; they will be discussed as follows:

Token Embeddings

The purpose of the Token Embeddings layer is to transform words into vector representations of a fixed dimension. Each word is represented as a 768-dimensional vector in the BERT case. The input text is first tokenized before being passed to the Token Embeddings layer. Additionally, Extra tokens are added at the start ([CLS]) and end ([SEP]) of the tokenized phrase. These tokens are intended to act as an input representation for the tasks of classification and to isolate a pair of input texts.

Segment Embedding

BERT can solve NLP tasks including text classification to give a pair of input texts. An instance of such an issue is the classification of two pieces of text, which are semantically similar. The pair of input texts are simply conjugated and fed into the introduced model.

Position Embeddings

The positional embeddings are used to describe the position of words in a sentence. These embeddings are added to control the transformer limitation, which is not able to capture the order information. Accordingly, the positional embeddings allow the BERT to understand the given input texts. To generate a single representation, these representations are summed up element-wise. This representation became the input representation which is passed to BERT's Encoder layer.

3.2.2 Encoder Layer

The Encoder layer consists of 12 blocks of Transformers and 12 heads of self-attention by taking an input of no more than 512 tokens of a sequence and generating the sequence representations. The representations may be a particular hidden state vector or a hidden state vector time-step sequence. Self-attention is used to learn the relation of any two tokens in a sentence clearly and take the sentence's inner structural details. The multi-head mechanism of self-attention is concerned in this paper. The scaled dot-product attention mechanism [6] is computed as shown in Eq. (1).

$$Attention(Q, K, V) = Softmax\left(\frac{QK^T}{\sqrt{d_K}}\right)V \tag{1}$$

where Q is the matrix of the query, K is the key matrix, V is the matrix of values, and d_k is the matrix dimensions of Q and K. Multi-head attention uses various linear projections to first linearly project the questions, keys, and values h times. The scaled dot-product attention is then carried out in parallel by the h projections. Finally, the results are conjugated and projected to produce a new representation once again. Formally, it is possible to express the multi-head attention as shown in Eqs. (2) and (3).

$$MultiHead(Q, K, V) = Conc(Head_1, .Head_n)W^o \tag{2}$$

$$Head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{3}$$

where matrices of weight are dimensioned correctly by W, the beauty of multi-head attention is the easy parallelization of the operation, which leads to reduced runtime. Each layer in our encoder and decoder includes a completely linked feedforward network in addition to the attention sub-layers, which is applied separately and identically to each position. It consists of two linear transformations triggered by a ReLU.

$$FFN = \max(0, X^{W1} + b1)W2 + b2 \tag{4}$$

The FNN output is transmitted as an input to the upper encoder layer [6]; this procedure is repeated until the output is sent to the last FNN by the uppermost encoder to the classification layer to predict the output. In NLP, the timing feature is considered a significant feature. Since the attention mechanism cannot extract the timing of the attribute, the transformer uses position embedding to add timing information, as shown in Eqs. (5) and (6):

$$PE(pos, 2i) = sin(pos/10000^{2i/d_{model}}) \tag{5}$$

$$PE(pos, 2i + i) = cos(pos/10000^{2i/d_{model}}) \tag{6}$$

where the pos is position and the dimension is i, a sinusoid corresponds to every positional encoding dimension of the. A geometrical progression from 2S to 10000 2S is generated by the wavelengths. This function is chosen as we hypothesized that the model might easily learn to participate in relative positions since for any fixed offset k, it is possible to represent PEpos k as a linear function of PEpos.

The sum of position embedding, type embedding, and word embedding is the input of BERT. In comparison to various language models, BERT can make full use of the information on the right and the left sides of words to acquire the best-distributed representation of words.

3.3 CNN

In this model, the pre-training BERT model is used to learn the word and sentences embedding. The embedding vector is the CNN input subsequently extracted from BERT. The CNN's overall structure is formed of Convolutional layer, Max pooling and fully connected layer are showed in Fig. 3.

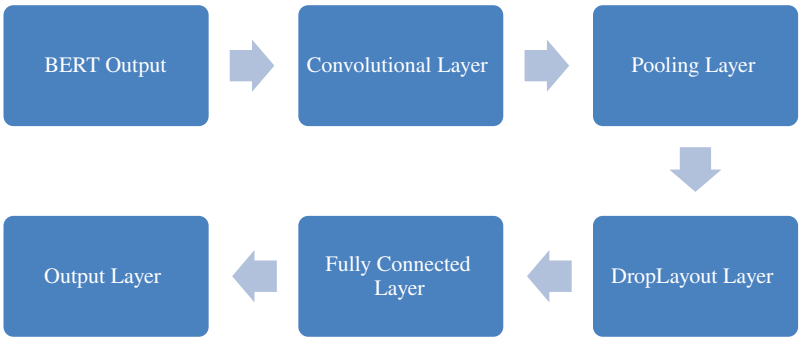

Figure 3: Architecture of the Convolutional Neural Network

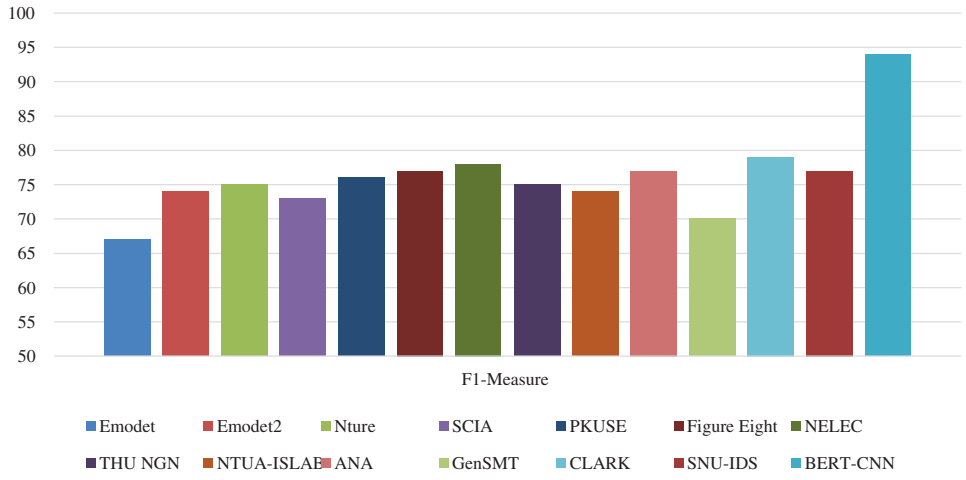

Figure 4: Comparing the BERT-CNN model with several recently proposed models using the semeval 2019 task 3 dataset

3.3.1. Convolution Layer

From the input matrix, layers must be extracted the higher-level features. Filters of different sizes should be added to get different types of features as many as reported in previous works [11]. The

width of each filter is set and processed as m and its height h as a hyper parameter. Given a filter $\omega \in Rh \times m$, a function Sj is created from a window of words and concepts [vi: vi+h−1] by:

$$Sj = g(\omega \cdot [vi : vi + h - 1] + b) \tag{7}$$

where, $b \in R$ is a bias term and g is a non-linear function. For convolution layers, we use ReLU as the activation function. To create a function map $s \in R^{n+k-h+1}$, the filter is extended to all possible windows of words and concepts in W. For different filters with different heights, this process can be repeated to improve the model's feature coverage.

3.3.2 Pooling Layer

After the Convolutional layer, the Max-pooling layer minimizes and down-samples the features in the feature map. A max-pooling operation is applied over each function map [47]. The idea is to select the highest value for each vector dimension to capture the most significant function. And then, the dropout technique is applied to reduce overfitting with the dropout value is 0.5. The pooling layer's output is calculated as shown in Eq. (8)

$$h^p = MaxPooling \quad (h^c|h, w) \tag{8}$$

where max-pooling layer output is represented by h^p, the sliding window height and width are represented by h and w respectively, and the output of the convolutional layer is represented by h^c .

3.3.3 Fully Connected and Output Layer

After extracting high-level features, the features are sent to the final layer. In this layer, the softmax function is used. The calculation of the softmax value is shown in Eq. (9):

$$p_i = \frac{e^j}{\sum^j e^j} \tag{9}$$

where Pi denotes the probability of the i^{th} class, e^i indicates the corresponding value of the output of i^{th} class and j denotes the total number of classes.

4 Experiments

This section provides the details of the experiment performed in this study and the discussion of the obtained results. The used datasets and a comparison with a state-of-art models are also reviewed. The results are shown in Tabs. 4–8.

4.1 Hyper-parameters

The experiment is implemented on the Google Colaboratory (Colab) that provides the Jupyter notebook environment and executes code in Python 3.6.9. The Colab supports the Tesla K80 GPU accelerator. The Keras front end runs on the TensorFlow backend. It enables fast experimentation of deep learning models by running code on the graph processing unit (GPU) and central processing unit (CPU). The classification performance of the models is evaluated by using the sklearn metrics.

In our system, the proposed method for emotion detection includes hidden layer size, number of layers, batch size, learning rate, and dropout. The hyper parameters are investigated with the best model classification effect. We use Adam optimizer with the Categorical cross-entropy loss function. Categorical cross-entropy enables our model to allocate the labels' independent probabilities; it is a requirement for problems with multi-label classification. As shown in the state of art papers, the experimental dataset split into a training set, validation, and testing set. Tab. 1 shows the optimum hyper-parameters obtained from repeated experiments.

Table 1: Hyper-parameters of the BERT-CNN model

Hyper-parameters	Values
Learning rate	4e-5
Loss Function	Categorical Cross-entropy
Optimizer	Adam
Batch size	16
Dropout	0.5
Convolutional size	3×3
Kernel sizes	[3–5]
Epochs	10

4.2 Dataset

The proposed model have been evaluated through two different datasets namely semeval 2019 task3 dataset [15] and ISEAR dataset [13]. Semeval 2019 dataset contains four classes of emotions: angry, happy, sad, and others. For data balance, the "others" label is removed from the dataset. The training, evaluation, and testing samples for each emotion category and the overall number of instances are shown in Tab. 2. ISEAR dataset, which consists of 7666 sentences labelled with respect to the following seven emotions: joy, anger, sadness, fear, shame, guilt and disgust. Tab. 3. presents the description of the ISEAR dataset.

Table 2: Description of the Semeval dataset

Dataset Split	Size	Happy	Sad	Angry	Others
Training	30160	4243	5463	5506	14948
Validation	2755	142	125	150	2338
Testing	5509	284	250	298	4677

Table 3: Description of the ISEAR dataset

Emotion labels	Quantity
Anger	1096
Disgust	1096
Sadness	1096
Joy	1094
Shame	1096
Guilt	1093
Fear	1095
Total	7666

Table 4: Comparing performances of the BERT-CNN model and several recently proposed models using semeval dataset

Model	F1	Model	F1
Emodet [16]	0.67	SNU_IDS [30]	0.77
Emodet2 [14]	0.74	THU NGN [23]	0.75
Nture [17]	0.74	NTUA-ISLAB [25]	0.74
SCIA [18]	0.73	ANA [27]	0.77
PKUSE [19]	0.75	ClARK [31]	0.79
Figure Eight [21]	0.76	GenSMT [29]	0.70
NELEC [22]	0.77		
BERT-CNN	**0.94**		

Table 5: Comparing performances of the BERT-CNN model and several recently proposed models for ISEAR dataset

Model	F1-Score
SVM [33]	0.55
RandomForest [33]	0.49
BiLSTM+ CNN+ Self-Attention +Fast Text [33]	0.63
BERT [13]	0.70
RoBERTa [13]	0.74
XLNET [13]	0.73
DistilBERT [13]	0.69
BERT- BiLSTM [12]	0.73
BERT-CNN	**0.76**

Table 6: Fine-grained emotion classification results for the test dataset based on the proposed model

Emotions	Precision	Recall	F-measure
Angry	0.91	0.97	0.94
Happy	0.98	0.96	0.97
Sad	0.95	0.90	0.92
Average total	0.946	0.943	0.943

Table 7: Comparing performances of the BERT-CNN model and several recently proposed models based on individual emotions scores using the test dataset

Emotions	Angry			Happy			Sad		
Model	P	R	F1	P	R	F1	P	R	F1
LIRMM [24]	0.72	0.80	0.76	0.72	0.70	0.71	0.82	0.77	0.80
SymantoResearch [26]	0.73	0.79	0.76	0.75	0.72	0.73	0.82	0.80	0.81
EPITA-ADAPT [20]	0.73	0.75	0.74	0.70	0.70	0.70	0.82	0.75	0.78
CAiRE_HKUST [28]	0.73	0.79	0.76	0.74	0.67	0.71	0.81	0.77	0.79
SINAI(BS) [32]	0.53	0.77	0.66	0.50	0.64	0.56	0.59	0.74	0.66
SINAI(BS-2) [32]	0.59	0.80	0.68	0.62	0.71	0.66	0.64	0.77	0.70
SINAI(SF) [32]	0.58	0.81	0.68	0.61	0.70	0.65	0.61	0.78	0.69
Bi-directional LSTM	0.47	0.78	0.59	0.51	0.58	0.54	0.51	0.76	0.61
CoAStaL [15]	0.68	0.82	0.75	0.72	0.66	0.69	0.74	0.77	0.75
BERT-CNN	**0.91**	**0.97**	**0.94**	**0.98**	**0.96**	**0.97**	**0.95**	**0.90**	**0.92**

4.3 Evaluation Measures

In this paper, standard metrics like precision, recall, accuracy, and F1 score are used for evaluation. Here, the number of positive sentences that are classified into emotion class correctly is indicated as True Positive (TP), while the number of negative sentences that are classified as negative into emotion class is indicated as True Negative (TN). The number of negative sentences classified as positive into emotion class is indicated as False Positive (FP), while the number of negative sentences that are classified as negative into emotion class is indicated as False Negative (FN). Precision, recall, accuracy, and F-measure is computed as shown in Eqs. (10)–(13).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{10}$$

$$Precision = \frac{TP}{TP + FP} \tag{11}$$

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

$$F - measure = \frac{2 * P * R}{P + R} \tag{13}$$

Table 8: Fine-grained emotion classification results for the ISEAR dataset based on the proposed model

Emotions	Precision	Recall	F-measure
Angry	0.66	0.68	0.67
Fear	0.88	0.80	0.84
Disgust	0.85	0.75	0.80
Guilt	0.66	0.77	0.71
Joy	0.90	0.86	0.88
Sadness	0.71	0.88	0.79
Shame	0.71	0.55	0.62
Average	0.77	0.76	0.76

4.4 Compared Algorithms

The BERT-CNN performance is compared to performances of the state-of-the-art models using two datasets mentioned in the previous sections. These models are Emotdet [16], EMODET 2 [14], Nture [17], SCIA [18], Coastal [15], PKUSE [19], EPITA-ADAPT [20], Figure Eight [21], NELEC [22], THU NGN [23], LIRMM [24], NTUA-ISLab [25], Syman to Research [26], ANA [27], CAiRE-HKUST [28], GenSMT [29], SNU_IDS [30], CLARK [31], and SINAI [32].

4.5 Results and Discussion

In this section, the performance results of the proposed model are presented for each class of fine-grained emotions on the testing dataset. It can be observed that the precision, recall, and f1-score of "happy" are high as compared to other emotions after handling the class imbalance as shown in Tab. 6. The BERT-CNN model attains new state-of-the-art results with 94.7% accuracy and 94.3% F1 measure on the semeval dataset.

The performance of the BERT-CNN model is assessed with respect to the baseline studies. A comparison study is carried out to compare the proposed model with the state-of-the-art models. Results are presented in Tab. 4 and Fig. 4. These results show that the proposed model outperforms the other methods in terms of its F-measure that is 94%.

Tab. 7 show a comparison of the proposed model with different state-of-the-art deep learning algorithms using three different emotions (e.g., angry, happy, and sad). The experimental results include precision, recall, and F1-Measure evaluation metrics for each emotion using different methods. The BERT-CNN model has achieved new state-of-the-art results on the happy emotion with 98% of precision, 96% of recall, and 97% of F1-measure, which are higher than all the compared models. While with the angry emotion it has achieved 91% of precision, 97% of recall, and 94%. With the sad emotion, it has achieved 95% of precision, 90% of recall, and 92% of F1-measure, which outperforms the results of the models compared.

Tab. 8. shows the performance of the proposed model on seven emotions (Angry, Shame, Joy, Guilt, Sadness, Fear and Disgust) as precision, Recall and F1-Score.

Tab. 5. and Fig. 5 show performance results of BERT-CNN model compared with other recent state of art models using ISEAR dataset. These results shows that the performance of proposed model outperformed the other models with an F1-Score 76%.

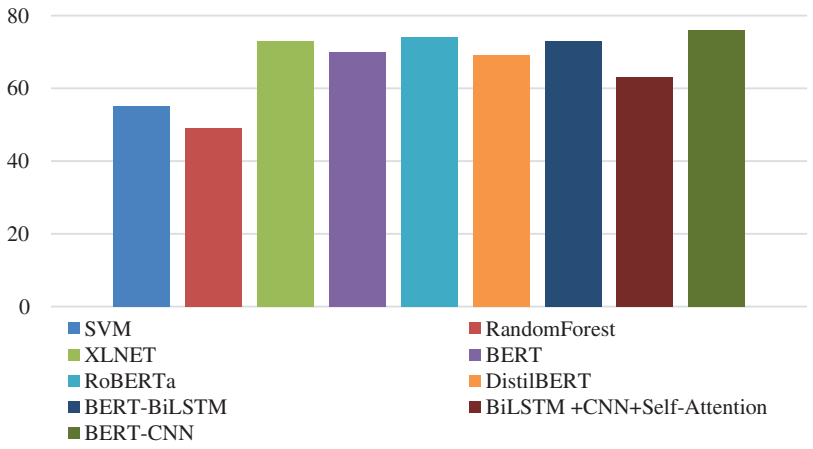

Figure 5: Comparing the BERT-CNN model with several recently proposed models using the ISEAR dataset

The appropriate choice of learning rate is important for the optimization of weights and offsets. If the learning rate is too large, it is easy to exceed the extreme point, which makes the system unstable. Moreover, provided the learning rate is too small, the training time is too long. The classification results of the proposed model at different learning rates are presented in Fig. 6.

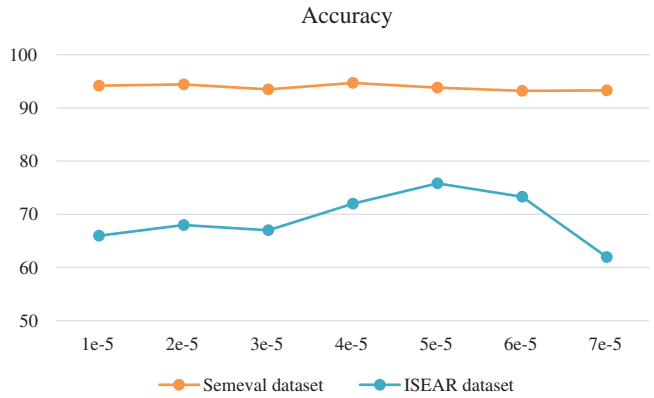

Figure 6: The impact of using different learning rates on the BERT-CNN model accuracy

Fig. 6 shows the results of testing the proposed model with various learning rates. It is reported that the highest accuracy is obtained with SemEval dataset when the learning rate is equal to 4e−5. However, the highest accuracy is obtained with ISEAR dataset when the learning rate is equal to 5e−5. It was also discovered that variations in learning rate value have a considerable impact on the presented model's performance.

Fig. 7 show the performance of the proposed model on both training and validation loss and accuracy.

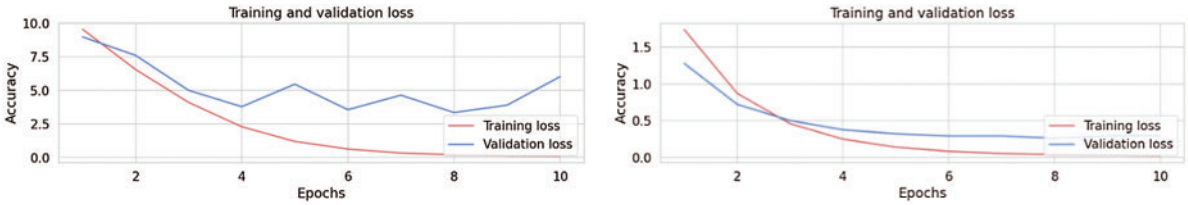

Figure 7: The results of accuracy and loss for training and validation for ISEAR dataset and semeval dataset

Figs. 8 and 9. shows the confusion matrix corresponding to evaluating the BERT-CNN model using the semeval dataset and ISEAR different motions.

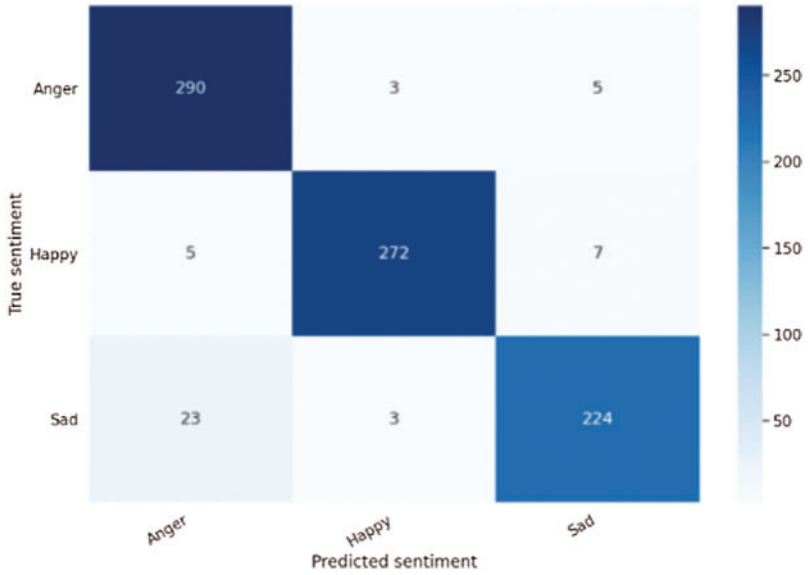

Figure 8: Confusion matrix of the BERT-CNN model on the semeval dataset

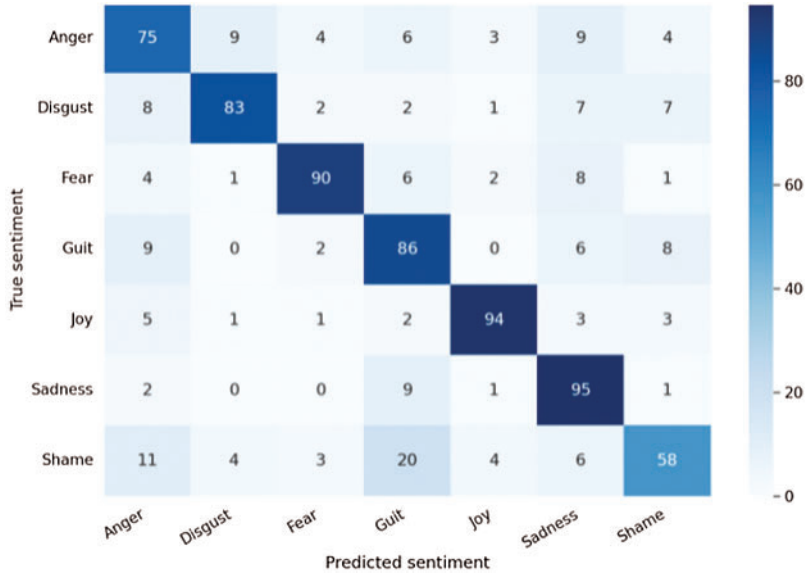

Figure 9: Confusion matrix of the BERT-CNN model on the ISEAR dataset

5 Conclusion and Future Work

In this paper, we introduced a new deep learning model, named BERT-CNN. This model is proposed to detect emotions from the text. This model is composed of the pre-trained BERT model combined with a convolutional neural network (CNN). It is used to detect emotions from text in the semeval2019 task3 dataset. This dataset contains multiple text that have three emotions, which are Happy, Sad and Angry. The BERT-CNN overcomes the state-of-the-art performance using four performance measures, which are Precision, Recall, F-measure and accuracy, compared to other models proposed in the literature using the semeval dataset.

In future work, we plan to improve the performance of our methods by replacing BERT with other pretrained transformer models like XLNet and RoBERTa. We also intend to extend the experiments using the Projection Attention Neural Network [48]. Textual emotion classification might be applied on a large dataset containing multiple languages. In addition, new methodology could be used to extract the reason that causes every emotion.

Acknowledgement: Authors thank those who contributed to write this article and give some valuable comments.

Funding Statement: The authors received no specific funding for this study.

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

References

[1] D. Haryadi and G. P. Kusuma, "Emotion detection in text using nested long short-term memory," *11480 (IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 10, pp. 6, 2019.

[2] K. Sailunaz, M. Dhaliwal, J. Rokne and R. Alhajj, "Emotion detection from text and speech: A survey," *Social Network Analysis and Mining*, vol. 8, pp. 1–26, 2018.
[3] Z. Jianqiang, G. Xiaolin and Z. Xuejun, "Deep convolution neural networks for twitter sentiment analysis," *IEEE Access*, vol. 6, pp. 23253–23260, 2018.
[4] M. Munikar, S. Shakya and A. Shrestha, "Fine-grained sentiment classification using BERT," *IEEE*, vol. 1, pp. 1–5, 2019.
[5] H. Huang, X. Y. Jing, F. Wu, Y. F. Yao *et al.,*"DCNN-Bigru text classification model based on BERT embedding," in *2019 IEEE Int. Conferences on Ubiquitous Computing & Communications (IUCC) and Data Science and Computational Intelligence (DSCI) and Smart Computing, Networking and Services (SmartCNS)*, ShenYang, China, pp. 632–637, 2019.
[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones *et al.,* "Attention is all you need," in *Advances in Neural Information Processing Systems*, Long beach, California, USA: Curran Associates Inc., pp. 5998–6008, 2017.
[7] W. Meng, Y. Wei, P. Liu, Z. Zhenfang and Y. Hongxia, "Aspect based sentiment analysis with feature enhanced attention CNN-biLSTM," *IEEE Access*, vol. 7, pp. 167240–167249, 2019.
[8] U. Naseem, I. Razzak, K. Musial and M. Imran, "Transformer based deep intelligent contextual embedding for twitter sentiment analysis," *Future Generation Computer Systems*, vol. 113, pp. 58–69, 2020.
[9] Z. Gao, A. Feng, X. Song and X. Wu, "Target-dependent sentiment classification with BERT," in *IEEE Access*, vol. 7, ShenYang, China: IEEE, pp. 154290–154299, 2019.
[10] J. Devlin, M. W. Chang, K. Lee and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," arXiv preprint arXiv:1810.04805, *2018*.
[11] J. Wang, Z. Wang, D. Zhang and J. Yan, "Combining knowledge with deep convolutional neural networks for short text classification," in *Combining Knowledge with Deep Convolutional Neural Networks for Short Text Classification*, Melbourne, Australia: International Joint Conferences on Artificial Intelligence, 2017.
[12] A. F. Adoma, N. M. Henry, W. Chen and N. Rubungo Andre, "Recognizing emotions from texts using a bert-based approach," in *2020 17th Int. Computer Conf. on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, Chengdu, China, pp. 62–66, 2020.
[13] A. F. Adoma, N. M. Henry and W. Chen, "Comparative analyses of bert, roberta, distilbert, and xlnet for text-based emotion recognition," in *2020 17th Int. Computer Conf. on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, Chengdu, China, pp. 117–121, 2020.
[14] H. Al-Omari, M. A. Abdullah and S. Shaikh, "Emodet2: emotion detection in English textual dialogue using bert and bilstm models," in *2020 11th Int. Conf. on Information and Communication Systems (ICICS)*, Irbid, Jordan, pp. 226–232, 2020.
[15] A. V. Gonzalez, V. P. B. Hansen, J. Bingel and A. Sogaard, "Coastal at semeval-2019 task 3: affect classification in dialogue using attentive bilstms," in *Proc. of the 13th Int. Workshop on Semantic Evaluation*, Minneapolis, Minnesota, USA, pp. 169–174, 2019.
[16] H. Al-Omari, M. Abdullah and N. Bassam, "Emodet at semeval-2019 task 3: emotion detection in text using deep learning," in *Proc. of the 13th Int. Workshop on Semantic Evaluation*, Minneapolis, Minnesota, USA, pp. 200–204, 2019.
[17] P. Zhong and C. Miao, "Ntuer at semeval-2019 task 3: Emotion classification with word and sentence representations in RCNN," arXiv preprint arXiv:1902.07867, 2019.
[18] Z. Rebiai, S. Andersen, A. Debrenne and V. Lafargue, "SCIA at semEval-2019 task 3: sentiment analysis in textual conversations using deep learning," in *Proc. of the 13th International Workshop on Semantic Evaluation*, Minneapolis, Minnesota, USA, pp. 297–301, 2019.
[19] L. Ma, L. Zhang, W. Ye and W. Hu, "PKUSE at semeval-2019 task 3: emotion detection with emotion-oriented neural attention network," in *Proc. of the 13th Int. Workshop on Semantic Evaluation*, Minneapolis, Minnesota, USA, pp. 287–291, 2019.
[20] A. Bouchekif, P. Joshi, L. Bouchekif and H. Afli, "EPITA-Adapt at semEval-2019 task 3: detecting emotions in textual conversations using deep learning models combination," in *Proc. of the 13th Int. Workshop on Semantic Evaluation*, Minneapolis, Minnesota, USA, pp. 215–2019, 2019.

[21] J. Xiao, "Figure eight at semEval-2019 task 3: ensemble of transfer learning methods for contextual emotion detection," in *Proc. of the 13th Int. Workshop on Semantic Evaluation*, Minneapolis, Minnesota, USA, pp. 220–224, 2019.
[22] P. Agrawal and A. Suri, "NELEC at semEval-2019 task 3: Think twice before going deep," arXiv preprint arXiv:1904.03223, 2019.
[23] S. Ge, T. Qi, C. Wu and Y. Huang, "THU NGN at semEval-2019 task 3: dialog emotion classification using attentional LSTM-cNN," in *Proc. of the 13th Int. Workshop on Semantic Evaluation*, Minneapolis, Minnesota, USA, pp. 340–344, 2019.
[24] W. Ragheb, J. Aze, S. Bringay and M. Servajean, "LIRMM-advanse at semEval-2019 Task 3: attentive conversation modeling for emotion detection and classification," *In SemEval: Semantic Evaluation in NAACL-HLT*, Minneapolis, Minnesota, USA: Association for computational Linguistics, pp. 251–255, 2019.
[25] R. A. Potamias and G. Siolas, "NTUA-Islab at semeval-2019 task 3: determining emotions in contextual conversations with deep learning," in *Proc. of the 13th Int. Workshop on Semantic Evaluation*, Minneapolis, Minnesota, USA, pp. 277–281, 2019.
[26] A. Basile, M. F. Salvador, N. Pawar, S. Stajner, M. C. Rios *et al.,* "Symantoresearch at semEval-2019 task 3: combined neural models for emotion classification in human-chatbot conversations," in *Proc. of the 13th Int. Workshop on Semantic Evaluation*, Minneapolis, Minnesota, USA, pp. 330–334, 2019.
[27] C. Huang, A. Trabelsi and O. R. Zaiane, "Ana at semeval-2019 task 3: Contextual emotion detection in conversations through hierarchical lstms and bert," arXiv preprint arXiv:1904.00132, 2019.
[28] G. I. Winata, A. Madotto, Z. Lin, J. Shin, Y. Xu *et al.,* "CAire-hKUST at semEval-2019 task 3: Hierarchical attention for dialogue emotion classification," arXiv preprint arXiv:1906.04041, 2019.
[29] D. Bogdan, "GenSMT at semeval-2019 task 3: contextual emotion detection in tweets using multi task generic approach," in *Proc. of the 13th Int. Workshop on Semantic Evaluation*, Minneapolis, Minnesota, USA, 2019.
[30] S. Bae, J. Choi and S. G. Lee, "SNU-Ids at semeval-2019 task 3: Addressing training-test class distribution mismatch in conversational classification," arXiv preprint arXiv:1903.02163, 2019.
[31] J. Cummings and J. Wilson, "CLARK at semEval-2019 task 3: exploring the role of context to identify emotion in a short conversation," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, pp. 159–163, 2019.
[32] F. M. Plaza-del-Arco, M. D. Molina-Gonzalez, M. T. M. Valdivia and L. A. U. Lopez "SINAI at semEval-2019 task 3: using affective features for emotion classification in textual conversations," in *Proc. of the 13th Int. Workshop on Semantic Evaluation*, pp. 307–311, 2019.
[33] M. Polignano, M. Gemmis, P. Basile and G. Semeraro, "A comparison of word-embeddings in emotion detection from text using bilstm, cnn and self-attention," in *Adjunct Publication of the 27th Conf. on User Modeling, Adaptation and Personalization*, Laranca Cyprus, pp. 63–68, 2019.
[34] M. Z. Asghar, F. Subhan, M. Imran, F. M. Kundi, S. Shamshirband *et al.,* "Performance evaluation of supervised machine learning techniques for efficient detection of emotions from online content," arXiv preprint arXiv: 1908.01587, 2019.
[35] M. Suhasini and B. Srinivasu, "Emotion detection framework for twitter data using supervised classifiers," in *Data Engineering and Communication Technology*, Singapore, Springer, pp. 565–576, 2020.
[36] C. Bhagat and D. Mane, "Text categorization using sentiment analysis," in *Proceeding of Int. Conf. on Computational Science and Applications*, Singapore, Springer, 2020.
[37] B. Gaind, V. Syal and S. Padgalwar, "Emotion detection and analysis on social media," *Global Journal of Engineering Science and Researches*, vol. 6, pp. 78–89, 2019.
[38] M. S. Rani and S. Subramanian, "Attention mechanism with gated recurrent unit using convolutional neural network for aspect level opinion mining," *Arabian Journal for Science and Engineering*, vol. 45, pp. 6157–6169, 2020.
[39] W. Ragheb, J. Aze, S. Bringay and M. Servajean, "Attention-based modeling for emotion detection and classification in textual conversations," arXiv preprint arXiv:1906.07020, 2019.

[40] Z. Yu, Y. Wang, Z. Liu and X. Cheng, "Emotionx-antenna: An emotion detector with residual GRU and text CNN," in *The 7th international workshop on NLP for social media (Social NLP) @ ICAI*, 2019.
[41] A. Chiorrini, C. Diamantini, A. Mircoli and D. Potena, "Emotion and sentiment analysis of tweets using BERT," in *EDBT/ICDT Workshops*, Nicosia, Cyprus, 2021.
[42] I. Sindhu, S. M. Daudpota, K. Badar, M. Bakhtyar, J. Baber *et al.,* "Aspect-based opinion mining on student's feedback for faculty teaching performance evaluation," *IEEE Access*, vol. 7, pp. 108729–108741, 2019.
[43] M. Polignano, P. Basile, M. de Gemmis and G. Semeraro, "A comparison of word-embeddings in emotion detection from text using bilstm, cnn and self-attention," in *Adjunct Publication of the 27th Conf. on User Modeling, Adaptation and Personalization*, Larnaca Cyprus, 2019.
[44] J. A. Gonzalez, L. F. Hurtado and F. Pla, "ELirf-uPV at semEval-2019 task 3: snapshot ensemble of hierarchical convolutional neural networks for contextual emotion detection," in *Proc. of the 13th Int. Workshop on Semantic Evaluation*, Minnesota, USA, pp. 195–199, 2019.
[45] J. A. Gonzalez, L. F. Hurtado and F. Pla, "Transformer based contextualization of pre-trained word embeddings for irony detection in twitter," *Information Processing & Management*, vol. 57, pp. 102262, 2020.
[46] H. Fei, Y. Ren and D. Ji, "Implicit objective network for emotion detection," in *CCF Int. Conf. on Natural Language Processing and Chinese Computing*, Springer, Cham, pp. 647–659, 2019.
[47] A. R. Abas, I. El-Henawy, H. Mohamed and A. Abdellatif, "Deep learning model for fine-grained aspect-based opinion mining," *IEEE Access*, vol. 8, pp. 128845–128855, 2020.
[48] P. Kaliamoorthi, S. Ravi and Z. Kozareva, "PRADO: Projection attention networks for document classification on-device," in *Proc. of the 2019 Conf. on Empirical Methods in Natural Language Processing and the 9th Int. Joint Conf. on Natural Language Processing (EMNLP-IJCNLP)*, pp. 5012–5021, 2019.