ISLAMIC UNIVERSITY OF TECHNOLOGY (IUT)

ORGNAIZATION OF ISLAMIC COOPERATION (OIC)

Thesis Title:

"Emotion Detection in Online Social Networks: Using Deep Learning Approach"

Authors:

Tareq Khaled 170041067

Mahamat Djibrine 170041072

Hafso Mohamat 170041080

Supervisor:

Lutfun Nahar Lota

Assistant Professor, Department of CSE

A thesis submitted to the Department of CSE in partial fulfilment of the requirements for the degree of B.Sc.

Department of Computer Science and Engineering (CSE)

Islamic University of Technology (IUT)

A Subsidiary organ of the Organization of Islamic Cooperation (OIC)

Academic Year: 2021-2022

DECLARATION:

This is to certify that the work presented in this thesis, titled, **“Emotion Detection in Online Social Networks: Using Deep Learning Approach”,** is the outcome of the investigation and research carried out by Tareq Khaled, Mahamat Djibrine and Hafso Mohamat, under the supervision of Assistant Professor Lutfun Nahar Lota. It is also declared that neither this thesis nor any part thereof has been submitted anywhere else for the award of any degree, diploma, or other qualifications. Information derived from the published or unpublished work of others has been acknowledged in the text and a list o references is given.

Authors:

Tareq Khaled A National of Yemen Student ID: 170041067 IUT – Dhaka	Mahamat Djibrine A National of Cameroon Student ID: 170041072 IUT – Dhaka	Hafso Mohamat A National of Somali Student ID: 170041080 IUT – Dhaka

Supervisor:

Lutfun Nahar Lota
Assistant Professor
Department of Computer Science and Engineering
Islamic University of Technology

ACKNOWLEDGEMENT:

It is an auspicious moment for us to submit our thesis work by which we are eventually going to end our Bachelor of Science study. In the beginning, we want to express our heartfelt gratitude to Almighty Allah for his blessings to bestow upon us which made it possible to complete this thesis research successfully. Without the mercy of Allah, we wouldn't be where we are right now. All thanks and praises be to Allah.

We would like to express our grateful appreciation to Lutfun Nahar Lota, Assistant Professor, Department of Computer Science and Engineering, Islamic University of Technology for being our advisor and mentor. Her motivation, suggestions, and insights for this thesis have been invaluable. Without her support and proper guidance, this thesis would not see the path of the proper itinerary of the research world. Her valuable opinion, time, and input provided throughout the thesis work, from the first phase of thesis topics introduction, research area selections, the proposition of algorithm, modification, and implementation helped us to do our thesis work properly.

We would also like to thanks Dr. Abu Raihan Moftofa Kamal, Professor, Head of Department of Computer Science and Engineering, Islamic University of Technology for his enormous help in directing us towards our goal and helping us in time of our need. We are grateful to him for his constant and energetic guidance, constructive criticism, and valuable advice.

TABLE OF CONTENTS:

LIST OF FIGURES:

<u>LIST OF TABLES:</u>

ABSTRACT:

Emotion recognition is one of the most difficult jobs in the Natural Language Processing (NLP) sector since it relies significantly on contextual information and mixed emotions in a sentence during the emotion detection process. Therefore, we propose two deep learning approaches CNN and Bi-LSTM, we built these two models on a dataset that contains six levels of emotions. The two models have proven to give good accuracy above 90% on this dataset. From that, we have decided to try them out on thirteen levels of emotions to see if we can still achieve reasonable performance on a high level of emotions.

1. INTRODUCTION

Emotion is one of a person's most basic cues. Emotions are distinct states of mind manifested as feelings such as "glad," "sad," "joy," and others. Nowadays, The community can freely express feelings, communicate, and share people's opinions, thoughts, views, and perspectives on local and international issues, matters, and topics via text, image, audio, and video posts on Online Social Media (OSM) platforms, with our concentration on text processing. Therefore, by studying the viewpoints, we will be able to see how people react to changes and what they are thinking. This analysis will help socialists and scholars think about what kind of actions they should do to alter society for the better. Emotion analysis research is evolving into a variety of applications, ranging from sentiment analysis of review data to the development of emotional interactive chatbots. In this paper, we propose two deep learning models, CNN and Bi-LSTM, and present the results of our tests on both the six and thirteen-level emotion datasets, as well as the problems we faced during the trials.

2. LITERATURE REVIEW:

We started with sentiment analysis, which aims to extract information from human language in order to analyze thoughts and sentiments and assign polarities such as positive, negative, or neutral. Emotion detection, on the other hand, seeks to identify more particular sentiment tones such as happy, sadness, depression, anxiety, and so on. We will firstly explore the work that is been done in regard to sentiment analysis then, move into emotions detection.

The purpose of this sentiment analysis was to determine customer satisfaction with e-commerce. The evaluation can then assist e-commerce businesses in focusing on increasing service and company quality, which will result in greater traffic, sales, and profitability. The assessment is not only been performed on e-commerce retailers but also on public events such as games and elections or on a public figure to positively change the society.

Meylan Wongkar and, Apriandy [1] worked on a sentiment analysis application for twitter analysis that was conducted on 2019 Republic of Indonesia presidential candidates; they have used Naïve Bayes method to classify the level of sentiments of the society toward the presidential candidates.

According to the findings, the positive sentiment polarity of the Jokowi-Ma'ruf Amin duo was 45.45% and the negative sentiment polarity was 54.55 percent, while the Prabowo-Sandiaga pair obtained a positive sentiment score of 44.32 percent and a negative sentiment score of 55.68 percent. The combined data was then tested against the training data for each presidential contender, yielding an accuracy of 80.90% 80.10%.

In this study, Rapid Miner has been used to compare the accuracy of the Naive Bayes, SVM, and K-Nearest Neighbor (K-NN) methods. The Nave Bayes accuracy value was 75.58 percent, the SVM accuracy value was 63.99 percent, and the K-NN accuracy value was 73.34 percent. From the results comparison, Naive Bayes out performed both SVM and K-NN.

In addition to that, Anjume Shakir and Jyoti Arora[2] worked on Sentiment Analysis of Twitter Data using KNN Classification Technique. This work is been done on a movie dataset of around 6000 tweets, the dataset was preprocessed and then passed into the K-NN classifier.

The KNN method improves the accuracy of binary, ternary, and multi-class sentiment analysis. KNN has also been shown to take less time to execute than other existing approaches.

Finally, Sheeba Naz, and Aditi Sharan [3] have done their research work on sentiment analysis using support vector machine(SVM). They began by selecting appropriate pre-processing methods to reduce any noise that the data may include, which is an important step toward boosting classification accuracy. Stop removing words, URLs, @, #, and other special characters Delete the numbers, Stemming refers to the preparation steps taken to sanitize the data.

The extraction of feature sets required for classifier training is the following stage. The feature selection and combination have a significant impact on the classifier's performance. The labeled data as well as the selected characteristics are sent into the machine-learning algorithm, which is then used to create the classifier model. The classifier gives labels for testing data (unlabeled tweets) in the final stage.

Furthermore, they have concentrated on the synthesized features to be able to target the tweets content only. they take the advantage of using n-gram model to build up the feature extraction and worked with uni-gram, bi-gram, tri-gram as well as their combination with three different weights that are applied to the social indicators, TF that is meant for term frequency, TF-IDF for term frequency-inverse document frequency that assess how important the word is to a document, and binary.

They initiate a sentiment score vector of tweets as an external feature alongside n-gram features with the goal of examining its impact on SVM classifier performance.

They used four distinct n-gram feature sets including three different weighting techniques in their tests. When opposed to other feature sets, the unigram features perform the best in terms of accurate results. They then combined an external feature with n-grams and found that the SVM classifier had better performance in terms of accuracy. Based on these research works, we have implemented all the three algorithms and their performance are as follows: For Naive Bayes, we were able to achieve the highest accuracy that stands to 94.80% follows by KNN with 93.80% accuracy and lastly 91% accuracy for SVM. To sum up the experiment, Naive Bayes classifier outperforms both KNN and SVM in predicting the sentiment.

However, researchers are getting more interested in analyzing emotions concerning the fact that The sentimental analysis primarily focuses on sentence-level polarity prediction, with no or little attention paid to the emotion that led to that categorization, and adds very little value to risk assessment, advertisement services, or recommendation systems.. Therefore, emotion detection has become a critical task in Online Social Networks (OSNs) and has been attracting more and more attention both in academics and industry.

In this paper, we first analyze the emotions using Anger, love, surprise, fear, joy, and sadness are among the twenty-four emotion types in Plutchik's Wheel of Emotions paradigm, each of which has three levels of intensity (for example, Terror is more intense than Fear; Apprehension is less intense than Fear). Later on, in this paper, we will be seeing the performance of the two models CNN and BiLSTM on the thirteen levels of emotions where the intensity of emotions is taken into consideration.

Machine learning researchers have built algorithms to understand emotion. The researchers in Travis [4] explained how we can take the advantage of transforming the emojis that the data content into their corresponding emotions which provide more semantic meaning to the sentence.

Inline to that, Illendula and Sheth [5], the researchers studied the effect of emojis and images. They fed the BiLSTM model with quick text embeddings and EmojiNet with the retrieved features from the photos, and they applied the Attention method. To detect emotions, Rosenthal and al [6] employed a multi-view ensemble technique. They used features like bag-of-words and word2vec to train the models. They used classic machine learning techniques such as Logistic Regression and Support Vector Machines.

EmoDet2 [7], determines the emotion and sentiment in English textual dialogue and classifies it into four categories (Happy, Sad, Angry, and Other), they have used the Semeval 2019 [8] dataset.

This paper uses the concept of EmoDet-BiLSTM which is the standard Recurrent Neural Network (RNN) [9] is distinguished from Feed-Forward Network with a memory. Long Short Term Memory (LSTM) is a type of RNN that consists of a memory cell, an input gate, an output gate, and a forget gate.

The Bidirectional Long Short-Term Memory (BiLSTM) [10] is the advanced form of LSTM in which the BiLSTM feeds the algorithm with the data once from beginning to the end, and once from the end to the beginning.

This allows the network to extract additional information from the data.. In Sea-Hai Park [11], they conducted a text emotion classification of Tweet data using the CNN learning algorithm.

The embedding layer was utilized to vectorize the text in this learning process, and GloVe was used as the initial embedding model. The values of neural network layers are then changed for emotion classification using the backpropagation technique. When the best classification performance is achieved, the embedding layer is extracted.

3. METHODOLOGIES:

In this section, we describe our working process of emotional recognition. Our method consists of data collection, preprocessing, data splitting, and finally fit the data into the respective models CNN and Bi-LSTM as shown in the below figure1.

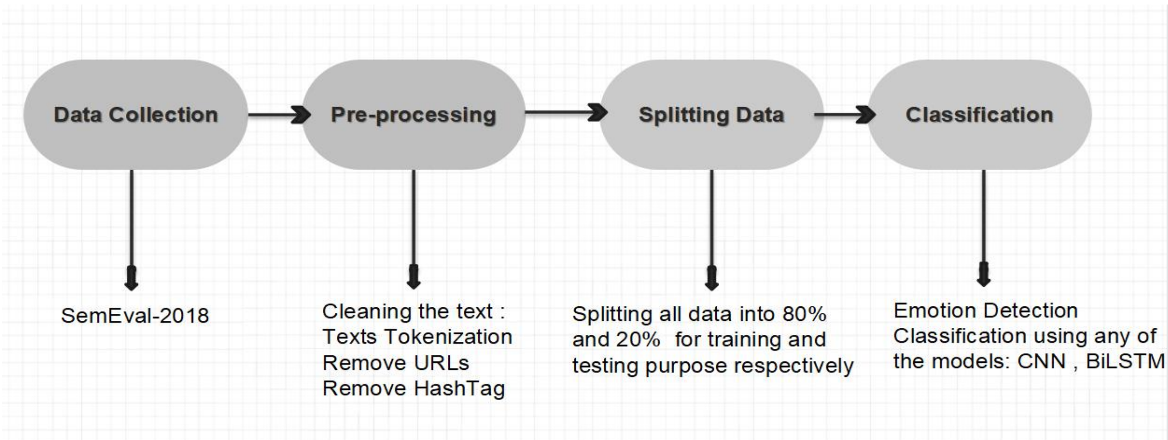


Figure1: Workflow

3.1 *Convolution Neural Network:*

A convolutional neural network consists of an input layer, hidden layers and an output layer. In our CNN architecture, the hidden layers include layers such ReLU (Rectified Linear Unit) as activation, embedding layer, pooling layer, dropout layer, and fully connected layer as described in figure 2.

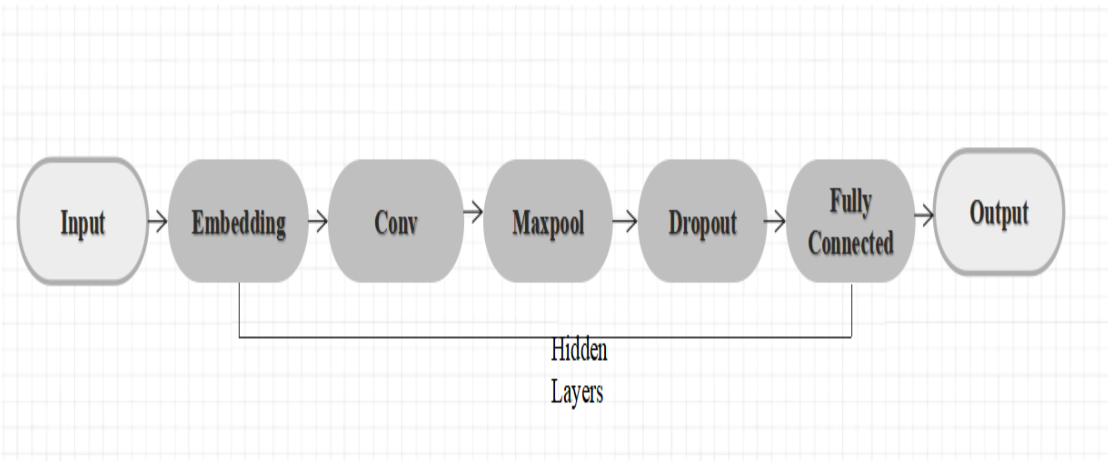


Figure 2: CNN Architecture

a) ***Embedding Layer:*** Word embedding is just a form representing words through vectors that successfully solve the drawback of derived using a one-hot encoding by somehow abstracting the context or high-level meaning of each word. The main takeaway here is that word embeddings are vectors that represent words

In way that similar meaning words have similar vectors. GloVe 6B 300d dataset is used as embedding layer.

b) ***Convolution Layer:*** The convolutional layer is made up of some filters whose parameters have to be trained. We have used 32 filters in which the set of filter size [2,3,5] is been used respectively. Filters are sliding through the entire input region and at each step where the scalar product between the input and the convolutional filter is been calculated and the result of the dot product produces the output of the convolutional layer.

c) ***Pooling Layer:*** We used one Max pool layer as it reduces the dimensions of data by combining the outputs of neuron clusters at one layer into a single neuron in the next layer. It is been done to reduce the computational requirements through the entire network.

d) ***Dropout Layer:*** This layer is meant to nullify certain random input values to generate a more general dataset and prevent the problem of overfitting and we have used the 0.1 dropout version.

e) **Fully Connected Layer:** The fully connected layer is the last layer of the convolutional neural network. A fully connected layer connects every neuron in one layer to every neuron in another layer and gives us the output for six classes.

3.2 Bi-directional Long Short Term Memory

Bi-directional long short-term memory is a sequence processing model that consists of two LSTMs, one taking the input in a forward direction, and the other in a backward direction [12]. Bi-LSTM model consists of an input layer, backward layer, forward layer, and an output layer. We used ReLU (Rectified Linear Unit) as activation layer in hidden layers. In the hidden layers, it includes other layers such as the embedding layer, pooling layer, dropout layer, and fully connected layer. The configuration of the layers remains the same as for the CNN. The architecture of BiLSTM is described in the figure below. As the architecture of the Bi-LSTM is build up based on the Backward and forward sequence encoding which implies that the prediction of the next word would not only depend on front words In the sentence but rather it depends also on the words that are in the backward. With that, Bi-LSTM could capture the feature from both side of words which indeed outperform the normal LSTM that capture only the front end words in the sentence to make the prediction.

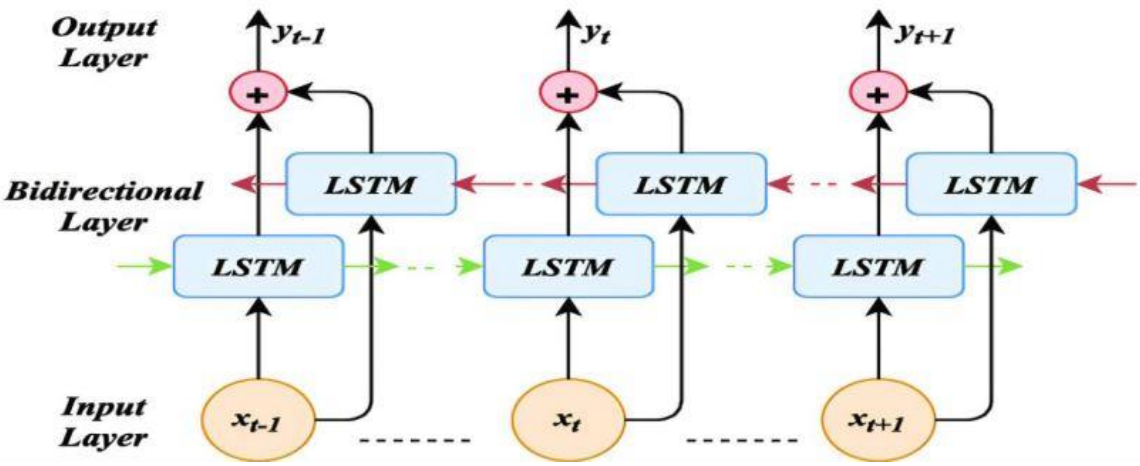


Figure 3: BiLSTM Architecture

4. EXPERIMENTS AND RESULTS:

4.1 **Experiment 1:** This experiment is been conducted on the **SemEval-2018** dataset that contains **six** levels of emotions.

A. Dataset Description:

The nature of the dataset is a Tweet Emotion dataset that is collected from kaggle with 20k size in which 80% of samples is used for training and 20% for testing purpose.

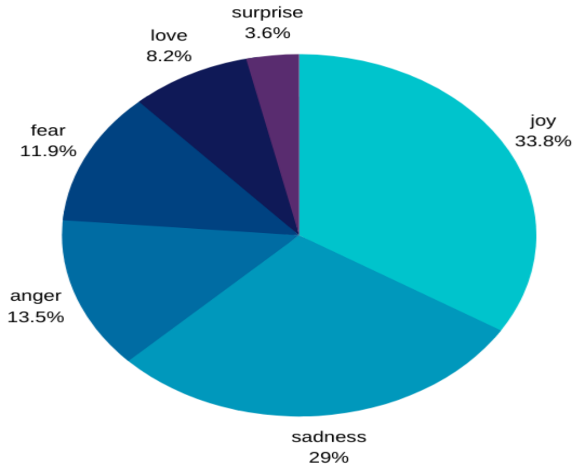


Figure 4: Data Statistics of Emotions of SemEval-2018 Dataset

Content	Sentiment
feel important share info experience thing	joy
Im grabbing minute post feel greedy wrong	Anger
i feel as confused about life as a teenager	fear
i feel romantic too	love
ive been taking or milligrams or times recomme...	surprise
i become overwhelmed and feel defeated	sadness

Table 1: Sample of the SemEval-2018 Dataset

B. Confusion Matrix:

A Confusion matrix is simply an N x N matrix that is used for evaluating the performance of a model classifier, where N is the number of target labels. The matrix compares the actual target values with those predicted by the machine learning model and in order word, the confusion matrix provides details summary of the prediction results of the classifier. The details matrix is shown the table below.

		Actual class		
		Positive	Negative	
Predicted class	Positive	**TP:** True Positive	**FP:** False Positive (Type I Error)	**Precision:** **TP** ----------------- **(TP + FP)**
	Negative	**FN:** False Negative (Type II Error)	**TN:** True Negative	**Negative Predictive Value:** **TN** ---------------------- **(TN+FN)**
		Recall or Sensitivity: **TP** ----------------- **(TP + FN)**	**Specificity:** **TN** ---------------------- **(TN + FP)**	**Accuracy:** **TP + TN** ----------------------------- **(TP + TN + FP + FN)**

Table 2: Confusion Matrix

Accuracy = (TP + TN) / (TP + FP +TN + FN)

Precision = TP / (TP + FP)

Recall = TP/ (TN + FN)

Specificity = TN/ (TN + FP)

F1 score = 2* (Precision * Recall) / (Precision + Recall) [13]

i. Confusion matrix of CNN:

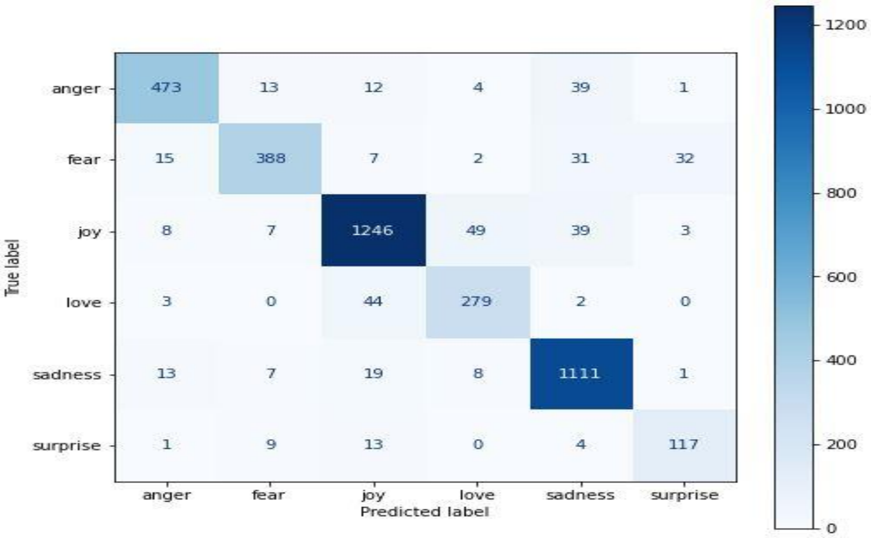


Figure 5: Confusion matrix of CNN

From the above confusion matrix, the CNN classifier were able to achieve significant prediction on the classes **sadness** and **joy** for which 96% of the **sadness** samples were correctly classified and 92% of the **joy** samples were correctly classified as well. Whereas, the classifier were able to correctly classify the rest of the classes with more 81% accuracy but below 90%. The out performance of the CNN model on both sadness and joy classes over the rest of the classes might be the fact that joy and sadness classes have the highest samples in the dataset that gives more features to the model to correctly classify them.

ii. Confusion matrix of Bi-LSTM:

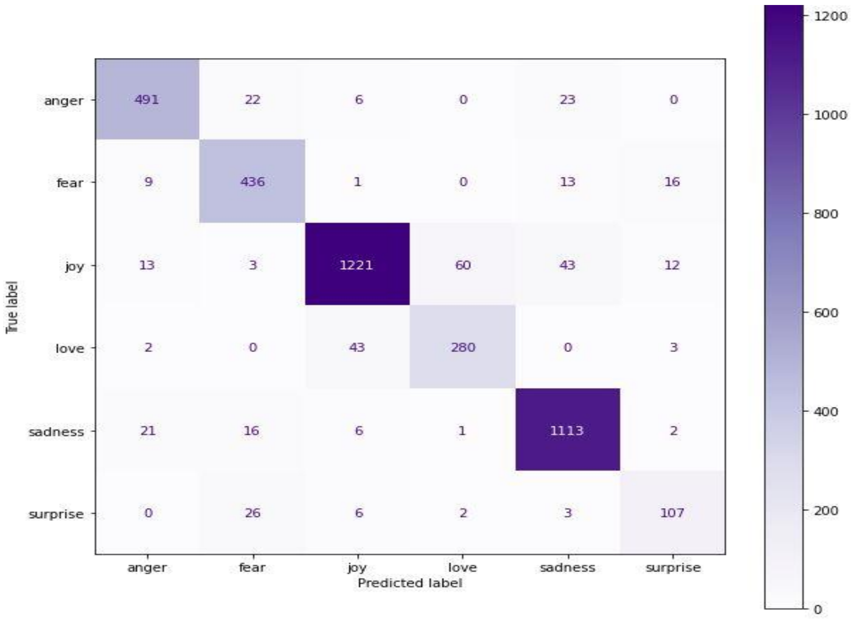


Figure 6: Confusion matrix of Bi-LSTM

The confusion matrix of Bi-LSTM looks similar to the CNN confusion matrix that maintains the correct prediction of both **joy** and **sadness** classes above 90% but the model achieved an accuracy below 80% for the class **surprise** that tells us the Bi-LSTM model tends to be less tolerable compared it to CNN model toward the class that has fewer samples in the dataset.

C. Results:

i. ***CNN Model Performance:***

Accuracy curve as well as the loss Curve of CNN is shown below respectively in figure 5 and figure 6.

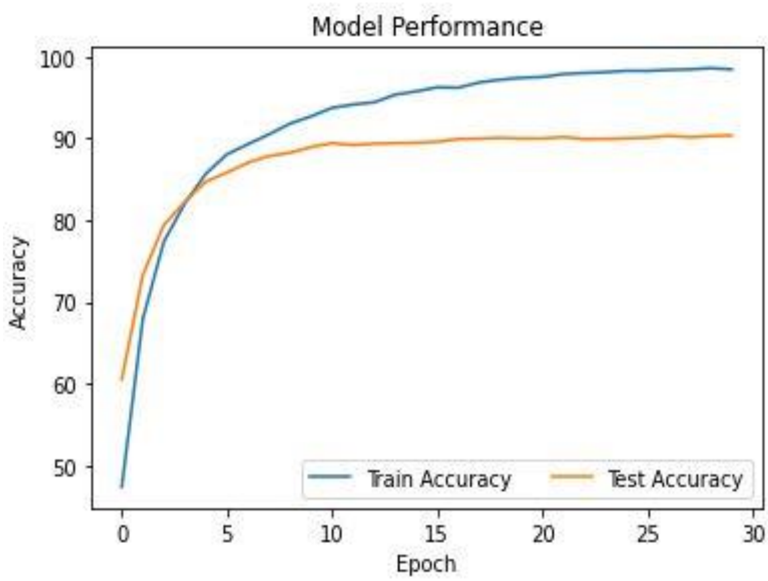


Figure 7: Accuracy Curve CNN

From the graph of the accuracy curve CNN, we can clearly observe that starting from epoch 1 up to epoch 5, both the train and test accuracy are increasing exponentially until they reached around 85% accuracy. After the epoch 5, the test accuracy started moving slower than the training accuracy until it reached a constant accuracy around 90% starting from epoch 15 onward.

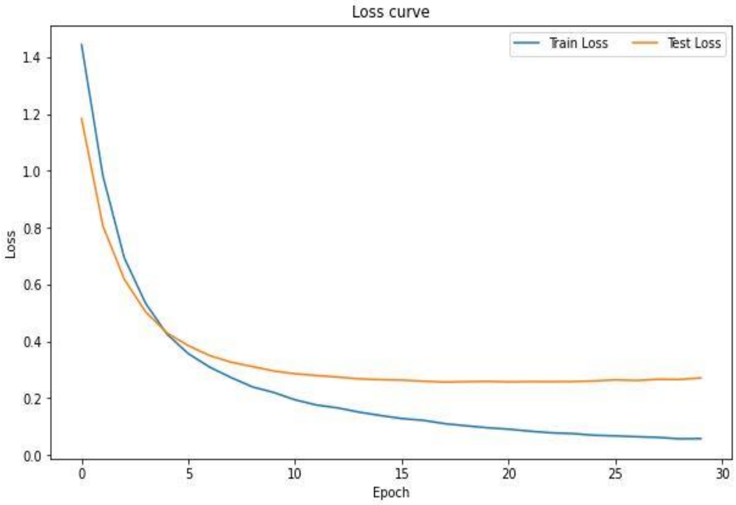


Figure 8: Loss Curve CNN

From the loss curve CNN, we can observe that starting from epoch 0 to epoch 1; the lost is at the pick as the model has not yet seen enough samples to learn from. Whereas, from epoch 1 up to epoch 5, the model has started seeing more samples that significantly decrease the lost as more features are been fed to the model and afterward, the lost continue to narrow down but this time slowly and it started becoming constant from epoch 15 at around 0.3 loss.

ii. ***Bi-LSTM Model Performance:***

Accuracy curve as well as the loss Curve of ***Bi-LSTM*** are shown respectively in figure 7 and figure 8 below:

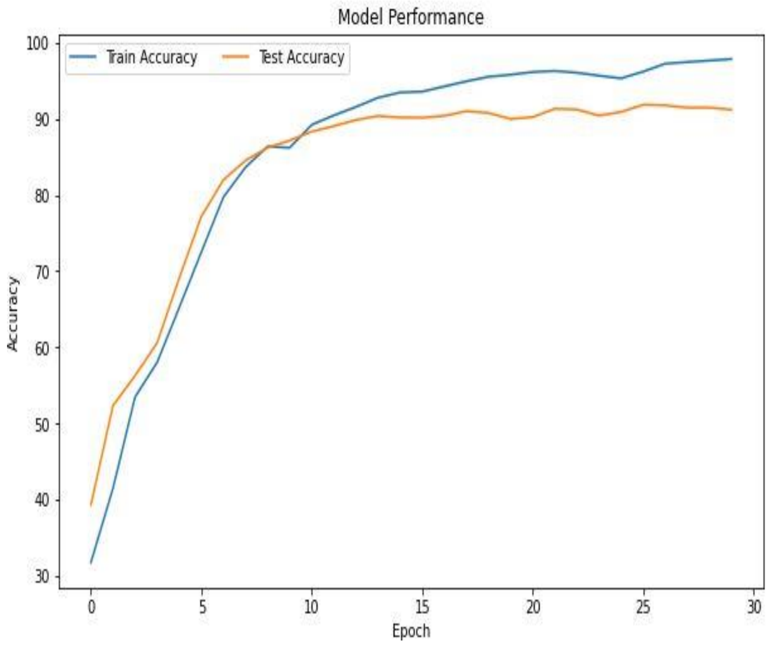


Figure 9: Accuracy Curve B-iLSTM

From the graph of the accuracy curve B-iLSTM, we can observe that starting from epoch 1 up to epoch 10, both the train and test accuracy are increasing exponentially until they reached around 87% accuracy. After the epoch 10, the test accuracy started moving slower than the training accuracy until it reached a constant accuracy around 90% starting from epoch 15 onward.

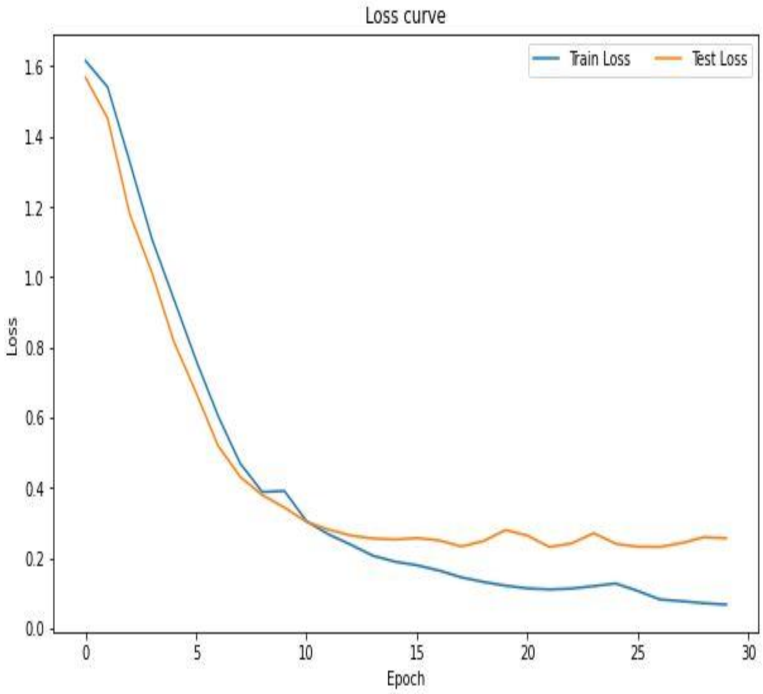


Figure 10: Loss Curve B-iLSTM

From the loss curve CNN, we can observe that starting from epoch 0 to epoch 1, the lost is at the pick as the model has not yet seen enough samples to learn from. Whereas, from epoch 1 up to epoch 10, the model has started seeing more samples that significantly decrease the lost as more features are been fed to the model and afterward, the lost continue to narrow down but this time slowly and it started becoming constant from epoch 15 at around 0.3 loss.

iii. *Precision for all Emotions:*

Emotions	**CNN**	**Bi-LSTM**
anger	92.203%	91.604%
fear	91.509%	86.680%
joy	92.916%	95.168%
love	81.579%	81.633%
sadness	90.620%	93.138%
surprise	75.974%	76.429%

Table 3: Precision for all of the emotions with respect to both Models

As we can see from this table that, the emotion that called joy it has the highest precision in both of implement models CNN and Bi-LSTM, for CNN model it achieved 92.916% and for Bi-LSTM model it also achieved 95.168%.

D. Model comparison:

Accuracy, Precision, Recall, F1- Score are evaluated after comparing between two models CNN and Bi-LSTM.

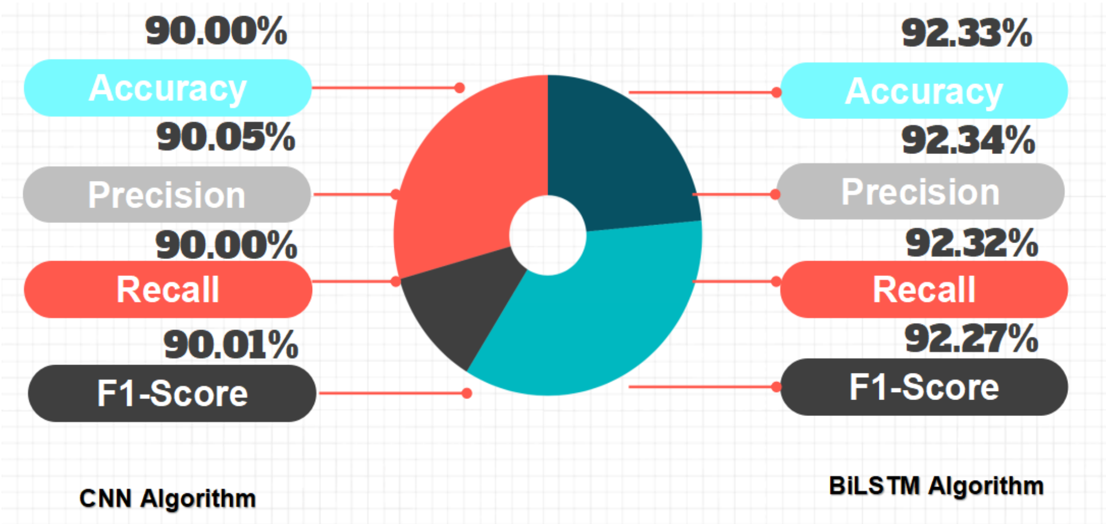


Figure 11; CNN vs. BiLSTM

i. **Conclusion on experiment 1:** The summary that can be drawn from this experiment is that both the two models CNN and Bi-LSTM performed well on the SemEval Dataset with a slight outperformance of Bi-LSTM over CNN. In this regard, we decided to push the boundary further with these two models and go for a large dataset with more levels of emotions that will introduce us to our second experiment.

4.2 **Experiment 2:** The second experiment is been conducted on the **text-emotions** dataset that contains **thirteen levels** of emotions.

A. **Dataset Description:**

Again the nature of the dataset is a Tweet Emotion dataset and it is collected from Kaggle as well. But this time twice larger in size than the previous dataset which results in a 40k size in which 80% of samples is used for training and 20% for testing purposes.

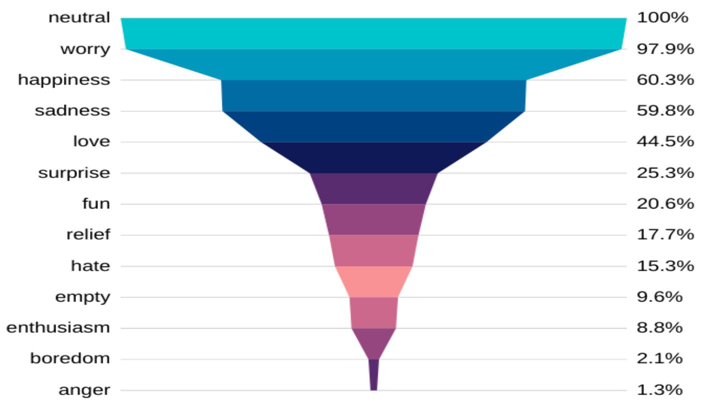


Figure 12: Data Statistics of Emotions of text-emotions Dataset

.

Content	**Sentiment**
mmm much better day... so far! it's still quit.	happiness
Im grabbing minute post feel greedy wrong	Anger
Choked on her retainers	worry
Happy Mother's Day to all the mommies out ther...	love
wants to hang out with friends SOON!	enthusiasm
feel important share info experience thing	sadness
Wondering why I'm awake at 7am,writing a new s...	fun
It is so annoying when she starts typing on her….	hate

Table 4: Sample of the text-emotions Dataset

B. Results:

We experienced a very low accuracy below **30 percent** for the first time we fed this dataset into the models CNN and Bi-LSTM. Then we observed that the dataset is much noisier than the previous one which requires further cleaning. While performing the intensive prepossessing we realize that we can take the advantage of the emoijs that the data contains and convert them into their relative semantic meaning which can eventually add some contextual information to the sentence and this may lead to an increase in the accuracy up to certain extend. But still, we could only manage to keep up the accuracy by **10 percent** as shown the figure 11 below.

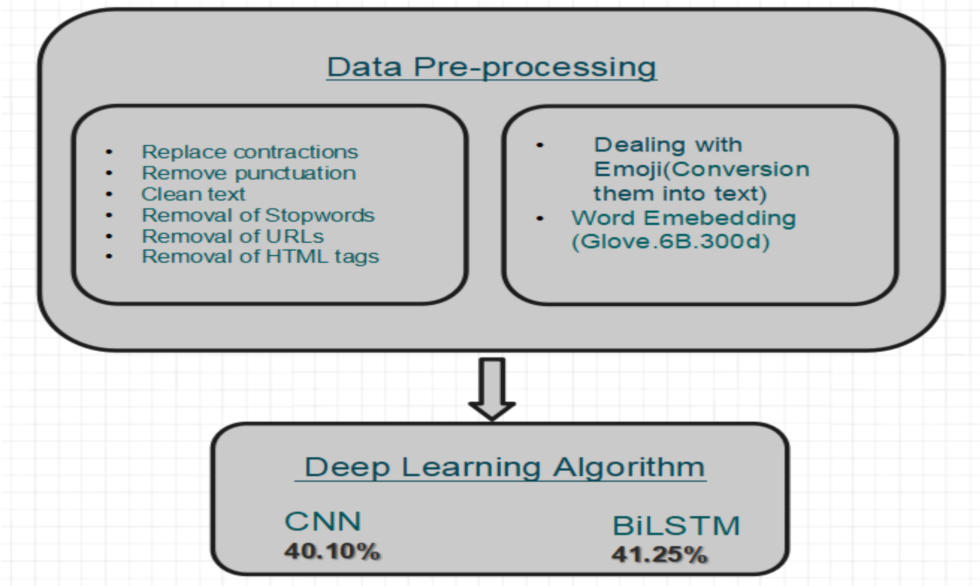
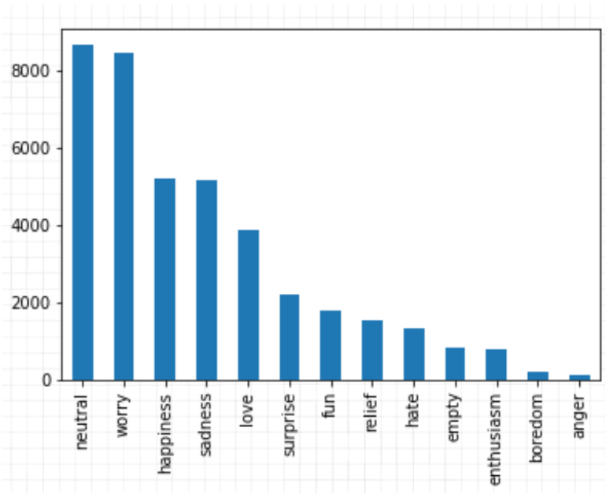


Figure 13: performance result on the text-emotions dataset

Fortunately while displaying the total count of each of the classes or labels as shown in figure 12 we observed that some of the classes are having more samples in the dataset than the others. Hence, the dataset in our hand is an imbalanced dataset that makes the model be biased towards the majority classes which will lead to diminishing the accuracy.



	CNN	Bi-LSTM
anger	NaN	NaN
boredom	0.000	NaN
empty	5.263	NaN
enthusiasm	0.000	NaN
fun	8.500	0.000
happiness	30.180	28.148
hate	33.065	40.000
love	37.976	43.559
neutral	35.625	34.331
relief	9.639	NaN
sadness	26.423	31.264
surprise	11.321	10.526
worry	34.981	33.914

Figure 14: Imbalance of the dataset description

To tackle this imbalance, we have used resampling, Augmentation and SMOTE Technique.

i. Resampling and Augmentation:

For both **Resampling** and **Augmentation** Techniques, their working principle is upon almost duplicating the sample of the minority classes to solve the problem of over-fitting. As they tend to duplicate the sample, then this will introduce the model to memorize rather learning therefore, the over-fit in which we could not achieve reasonable accuracy with these two techniques.

ii. **SMOTE:**

SMOTE is also known as Synthetic Minority Over-Sampling Technique, it is an oversampling technique that creates new minority class synthetic samples. Therefore, for our imbalanced dataset, first SMOTE is applied to create new synthetic minority samples to get a balanced distribution as shown in the Figure 13.

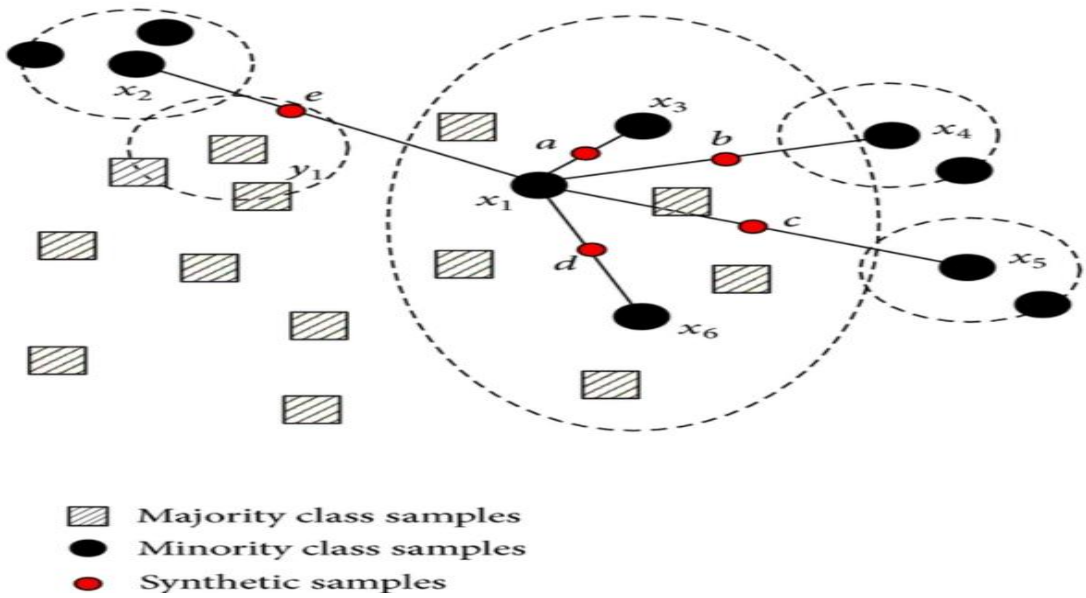


Figure 15: Visualization of SMOTE Technique

Simple is what smote does. For each of the samples in the class, it first determines the n-nearest neighbors in the minority class. Then it creates random locations on the lines by drawing a line between the neighbors. It determines the 5 closest neighbors to the sample points from the above image. Then a line is drawn through each of them. Then, it try to create some samples on the line with class that equal to minority class. As a result, this approach aids in overcoming the problem of over-fitting caused by random oversampling.

After implementing the SMOTE Technique on our training dataset, we are able to keep up the accuracy by over **10%** as shown below.

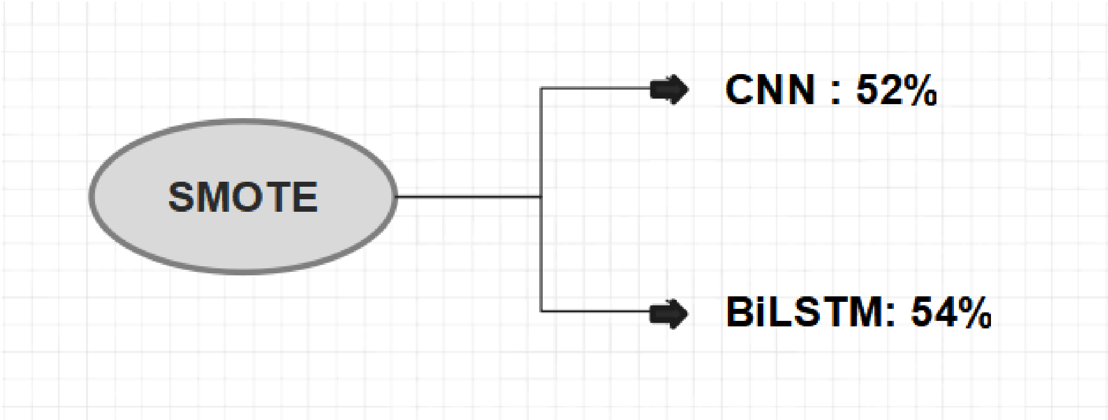


Figure 16: SMOTE technique performance

iii. **Conclusion on experiment 2:** In the first place, we have seen the impact of implementing the techniques **Resampling** and **Augmentation** to solve the problem of imbalance but as they tend to duplicate the same sample we could not achieve good performance with these two techniques but **SMOTE** Technique we are able to keep up the **accuracy** over **10%.** we could not achieve **high accuracy** with the **SMOTE technique** because of the following reasons:

1. As we know while we are generating synthetic examples, actually **SMOTE** does not take into consideration the neighboring examples from other classes. And this can result in increasing the overlapping of the classes and can introduce additional noise.

2. **SMOTE** is not very effective for the kind of data that we are dealing with because of the large size of our dataset. In this connection, still, there is a huge room for improvement starting from handling the imbalance dataset through further preprocessing, extracting samples

from twitter for the classes which are under samples and work on optimizing the two models.

5. CONCLUSION:

In this research, we built two deep learning models CNN and Bi-LSTM to identifier human emotional states. With both models, we were able to achieve above **90%** accuracy on the SemEval-2018 dataset but we could not achieve reasonable accuracy on the text-emotions dataset which tends to be much noisy and suffers from imbalance, we have seen some techniques such as resampling, Augmentation and SMOTE Technique. Out of these techniques, we were able to keep up the accuracy with the SMOTE Technique but still the accuracy could be further improved.

6. FUTURE WORK:

From our experiment 2, we are still working forward to improve the accuracy of the text-emotions dataset. We are currently working hard on techniques such as SMOTE and Augmentation to tackle the imbalanced data. In parallel to that, we are working on optimizing both CNN and Bi-LSTM to accurately predict the emotions.

REFERENCES:

[1] M. Wongkar and A. Angdresey, "Sentiment Analysis Using Naive Bayes Algorithm Of The Data Crawler: Twitter," 2019 Fourth International Conference on Informatics and Computing (ICIC), 2019, pp. 1-5, doi: 10.1109/ICIC47613.2019.8985884.

[2] A.Shakir and Jyoti Arora, “Sentiment Analysis of Twitter Data using KNN Classification Technique,”IJSRD - International Journal for Scientific Research & Development| Vol. 6, Issue 03, 2018 | ISSN (online): 2321-0613

[3] S. Naz, A. Sharan and N. Malik, "Sentiment Classification on Twitter Data Using Support Vector Machine," 2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI), 2018, pp. 676-679, doi: 10.1109/WI.2018.00-13.

[4] T. LeCompte and J. Chen, “Sentiment analysis of tweets including emoji data,” in 2017 International Conference on Computational Science and Computational Intelligence (CSCI). IEEE, 2017, pp. 793–798.

[5] A. Illendula and A. Sheth, “Multimodal emotion classification,” in Companion Proceedings of The 2019 World Wide Web Conference, 2019, pp. 439–449.

[6] S. Rosenthal, N. Farra, and P. Nakov, “Semeval-2017 task 4: Sentiment

analysis in twitter," arXiv preprint arXiv:1912.00741, 2019.

[7] H. Al-Omari, M. A. Abdullah, and S. Shaikh, "Emodet2: Emotion detection in english textual dialogue using bert and bilstm models," in 2020 11th International Conference on Information and Communication Systems (ICICS). IEEE, 2020, pp. 226–232.

[8] A. Chatterjee, K. N. Narahari, M. Joshi, and P. Agrawal, "Semeval-2019 task 3: Emocontext contextual emotion detection in text," in Proceedings of the 13th international workshop on semantic evaluation, 2019, pp. 39–48.

[9] T. Mikolov, M. Karafi ́at, L. Burget, J. Cernock `y, and S. Khudanpur, "Recurrent neural network based language model." in Interspeech, vol. 2, no. 3. Makuhari, 2010, pp. 1045–1048.

[10] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," IEEE transactions on Signal Processing, vol. 45, no. 11, pp. 2673–2681, 1997.

[11] S.-H. Park, B.-C. Bae, and Y.-G. Cheong, "Emotion recognition from text stories using an emotion embedding model," in 2020 IEEE international conference on big data and smart computing (BigComp). IEEE, 2020, pp. 579–583.

[12] "Bi-directional long short term memory description." https://paperswithcode.com/method/bilstm

[13] https://cedar.buffalo.edu/~srihari/CSE676/11.1%20PerformMetrics.pdf