Emotion-Text Classification Using Deep Learning Models

Initial Project Proposal

Course: CSCE 4604
By: Mona Ahmad Kamel - 900191833
Habeeba Hossam - 900191525
Under Supervision: Dr. Moustafa Youssef

Problem Statement

People's emotions represent a vast part of their personality, and can be utilized to understand and predict their actions. Expressing their emotions would reflect their thoughts and feelings, even the subconscious ones ranging from excitement and pride about a specific achievement to anger, depression, and even suicidal thoughts. Writing is one of the vital tools of expressing emotions. Consequently, it has become of uttermost importance to detect the emotion that one expresses using their texts as it would help in understanding people's perspectives, which in return increase people's empathy, identify any angry or violent texts, and it can even help in preventing violent actions or suicidal attempts.

Motivation

As previously mentioned, texts and writing are highly utilized by people in expressing their thoughts and emotions. This encompasses both manually writing journals or utilizing the growing social media platforms such as posting Tweets. Since technology has been on an ongoing breakthrough, Tweets and posts have become easily accessible and are abundant. There have been countless datasets that contain tweets, posts, paragraphs or journal/diary entries that have been labeled with various emotions. Therefore, classifying emotions using texts has become a more feasible, yet crucial problem. Our motivation is to utilize such vast online data to build an effective deep learning model that can aid in solving the problems mentioned above to reduce any negative action that can take place in the future.

Input/Output Examples

Since the aim is to use text for emotion classification, the input can be any text that would reflect emotional perspectives such as journals, diaries, poems, messages, dialogues, Tweets, and online posts on several social platforms such as Facebook and Reddit. The output should be the classification of the corresponding text. Therefore, it is crucial to recognize the various emotion taxonomies. There are three main taxonomies that are used in most papers. The first and the most common one is the Ekman's emotion taxonomy which consists of six main emotions: joy, anger, fear, sadness, disgust, and surprise [1]. By sentiment grouping, emotions usually have 3 or 4 categories which are positive, negative, neutral, and ambiguous if the emotion cannot be distinctly classified. Lastly, another taxonomy was proposed to present a more realistic classification of human's emotions with 27 categories, which can be found in the GoEmotions dataset [1].

Regardless of the taxonomy used in the data, the output should be a single label for each record. An illustrative example is shown in **Figure 01**. While the example uses a Facebook post, other types of texts, such as the one aforementioned, are also utilized.

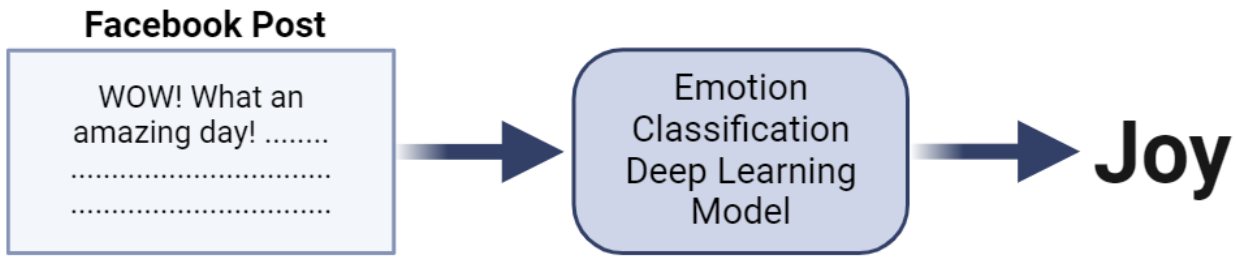


Figure 01. An illustration of the input and output of the emotion classification model

Survey of Evaluation Metrics

Evaluation metrics are measures used to evaluate the performance of a model. While various evaluation metrics exist, it is important to identify the useful ones based on the problem the model is aiming to solve. Regarding the emotion classification using text, some of the most commonly used metrics are the F1-score, recall, precision, and accuracy. Other performance metrics that are sometimes used for sentiment analysis and emotion classification is the confusion matrix, which shows the true and predicted values. However, in order to understand the importance of each metric, we need to define four main terms.

1. True Positive (TP): Number of correctly classified data points as positive, which means that the actual and predicted values are the same.
2. True Negative (TN): Number of correctly classified data points as negative.
3. False Positive (FP): Number of incorrectly classified data points as positive. For example, if a data point was truly “joy”, but it got predicted as “sad”, then when calculating the FP of “sad”, this point will be counted among the false positive ones.
4. False Negative (FN): Number of incorrectly classified data points as negative.

It is important to shed light on the fact that in multi-classification problems, these values are calculated for each class. To further elaborate, consider a dataset that uses Ekman's taxonomy. Since there are 6 classes, then these four values are calculated for each of the six classes. After defining the main terms, the evaluation metrics can be elaborated as follows:

1. **Confusion Matrix:**
   The confusion matrix is a visualization of the four values defined before, which are TP, TN, FP, and FN. All values are visualized in a matrix where the labels found beside the rows represent the predicted values, and the labels found at the top of each column represent the actual/true values.
2. **Recall:**
   The recall, which is also known as the **sensitivity** or true positive rate (TPR), is a metric that is used to evaluate the ability of the model to get the relevant results for a specific class. This is achieved by calculating the proportion of the correctly classified positive

values out of all the positive observations (whether they are correctly classified or not), which is shown in **Equation 1**. Therefore, a high recall would indicate that the model can correctly predict most of the positive observations of a class.

$$Recall = \frac{TP}{TP+FN}$$ **"Equation 1"**

3. **Precision:**
   Precision is the metric that calculates the proportion of correctly positive predicted instances from all predicted positive instances, whether they are correct or not. This can be shown in **Equation 2.** A high precision means that the model can correctly predict the positive instances of the desired class.

   $$Precision = \frac{TP}{TP+FP}$$ **"Equation 2"**

4. **F1-Score:**
   F1-score combines between the recall and precision. It attempts to balance between those two metrics. It is usually used when both metrics are important and when there is imbalance in the data. Since text-based emotion classification datasets usually have imbalanced classes, the F1-score is the most commonly used and the most reliable metric. The way the F1-score performs the trade off between the precision and recall is shown in **Equation 3.**

   $$F1\ Score = 2 * \frac{Precision * Recall}{Precision + Recall}$$ **"Equation 3"**

5. **Accuracy:**
   Accuracy is the metric used to evaluate the overall performance of the model in identifying data correctly as it measures the proportion of correctly predicted values, whether positive or negative, out of all predictions as shown in **Equation 4.** However, a main disadvantage of the accuracy metric is that it could be misunderstood in the case of imbalanced data. For instance, if there are two classes "normal" and "abnormal", and 90% of the data is normal, then even if most or all abnormal data points are incorrectly classified, the model will still have a high accuracy. Therefore, in the case of imbalanced data, like many of the emotion classification datasets, accuracy is not used. However, for the few papers that used the accuracy, they still used recall, precision, and/or F1-score as evaluation metrics to have more reliable results.

   $$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$ **"Equation 4"**

Current State of The Art Results

As mentioned above, while there are several evaluation metrics, the F1-score is the most reliable metric. Consequently, the current state-of-the-art paper was chosen to be the one with the highest F1-score. "An Optimized Deep Learning Model for Emotion Classification in Tweet" paper, which was published in 2021, has resulted in the highest F1-score as it was 87.7% [4]. It also has a very high precision of 90%. The details of the model architecture are discussed in the survey developed below.

Survey of Available Datasets

1. Sentiment140 Dataset

Link: https://www.kaggle.com/datasets/kazanova/sentiment140
Dataset Storage Size: 238.8 MB
Data Set Size and Split Information: There are 1600000 observations. The split of training, validation and testing was not explicitly mentioned.
Description: The dataset has Tweets collected using a Twitter API. It has several attributes such as the text itself, the date where the Tweet was published, the query if present, the user of the Twitter account, and the label. The label is an integer and is one of three possible options which represent three sentiments: negative (0), neutral (2), and positive (4). However, there were no Tweets labeled as neutral, so 2 will not be present as a label in the dataset.
Example: The input of the model will be all attributes except the label, which is the sentiment attribute, and the model should predict a sentiment for each Tweet as an output.

text string · lengths	date string · lengths	user string · lengths	sentiment int32	query string · classes
@switchfoot http://twitpic.com/2y1zl - Awww, that's a bummer. You shoulda got David Carr of Third Day to do it. ;D	Mon Apr 06 22:19:45 PDT 2009	_TheSpecialOne_	0	NO_QUERY
is upset that he can't update his Facebook by texting it... and might cry as a result School today also. Blah!	Mon Apr 06 22:19:49 PDT 2009	scotthamilton	0	NO_QUERY
@Kenichan I dived many times for the ball. Managed to save 50% The rest go out of bounds	Mon Apr 06 22:19:53 PDT 2009	mattycus	0	NO_QUERY
my whole body feels itchy and like its on fire	Mon Apr 06 22:19:57 PDT 2009	ElleCTF	0	NO_QUERY
@nationwideclass no, it's not behaving at all. i'm mad. why am i here? because I can't see you all over there.	Mon Apr 06 22:19:57 PDT 2009	Karoli	0	NO_QUERY

Figure 02. An example of the Sentiment140 dataset

2. ISEAR Dataset

Link: https://github.com/PoorvaRane/Emotion-Detector/blob/master/ISEAR.csv
Dataset Storage Size: 932 KB
Data Set Size and Split Information: There are 7666 observations in this dataset. The split of training, validation, and testing is not explicitly mentioned.
Description: The dataset consists of statements that reflect various emotions. These data were obtained from 1096 different participants who were asked about seven emotions, which are: anger, disgust, fear, sadness, shame, joy, and guilt.
Example: These are the first 14 observations of the dataset. As shown, there are no column headers, so the first step will be adding two column names, which can be "Label" and "Text" respectively. The dataset will be split into training, validation, and testing. Afterwards, the model will be trained by inserting the text into the model after preprocessing and the model will predict the corresponding emotion. It will then be validated and tested using the other two data groups.

	A	B
1	joy	On days when I feel close to my partner and other friends.
2	fear	Every time I imagine that someone I love or I could contact a
3	anger	When I had been obviously unjustly treated and had no possibility
4	sadness	When I think about the short time that we live and relate it to
5	disgust	At a gathering I found myself involuntarily sitting next to two
6	shame	When I realized that I was directing the feelings of discontent
7	guilt	I feel guilty when when I realize that I consider material things
8	joy	After my girlfriend had taken her exam we went to her parent's
9	fear	When, for the first time I realized the meaning of death.
10	anger	When a car is overtaking another and I am forced to drive off the
11	sadness	When I recently thought about the hard work it takes to study, and
12	disgust	When I found a bristle in the liver paste tube.
13	shame	When I was tired and unmotivated, I shouted at my girlfriend and
14	guilt	When I think that I do not study enough. After the weekend I

Figure 03. An example of the ISEAR dataset

3. CARER Dataset

Link: https://www.kaggle.com/datasets/parulpandey/emotion-dataset

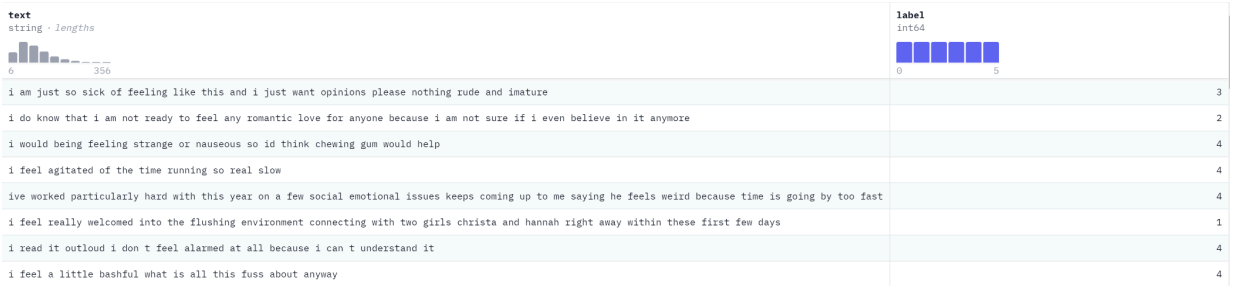
Dataset Storage Size: 1.92 MB

Data Set Size and Split Information: There are 20000 observations, where the data is split into 16000, 2000, and 2000 rows for training, validation and testing data. respectively.

Description: The dataset represents English Tweets that are classified by Ekman's taxonomy into six main emotions, where each emotion is represented by an integer. Below are the mappings of the emotions:

0: sadness
1: joy
2: love
3: anger
4: fear
5: surprise

Example: Below is an example of the training part of the dataset, showing 8 records. The input will be the text, which will be evaluated by the deep learning model, and the output will be 1 of the six possible integers, which correspond to one of the six emotions mentioned above.



text string · lengths 6 356	label int64 0 5
i am just so sick of feeling like this and i just want opinions please nothing rude and imature	3
i do know that i am not ready to feel any romantic love for anyone because i am not sure if i even believe in it anymore	2
i would being feeling strange or nauseous so id think chewing gum would help	4
i feel agitated of the time running so real slow	4
ive worked particularly hard with this year on a few social emotional issues keeps coming up to me saying he feels weird because time is going by too fast	4
i feel really welcomed into the flushing environment connecting with two girls christa and hannah right away within these first few days	1
i read it outloud i don t feel alarmed at all because i can t understand it	4
i feel a little bashful what is all this fuss about anyway	4

Figure 04. An example of the CARER dataset

4. WASSA-2017 Dataset

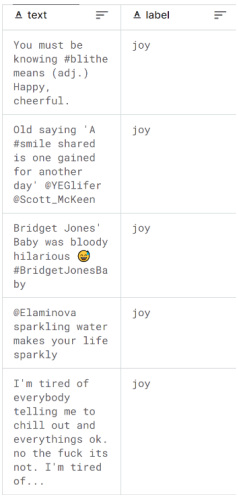
Link: https://www.kaggle.com/datasets/anjaneyatripathi/emotion-classification-nlp

Dataset Storage Size: 733.39 KB

Data Set Size and Split Information: There are 7006 rows that are split into training, validation, and testing with 3565, 342, and 3099 values respectively.

Description: This dataset represents 7006 Tweets and their corresponding labels where there are four emotions, which are: joy, sadness, anger, and fear.

Example: These are samples of the WASSA 2017 dataset. The input of the model is the text after any preprocessing steps and the output is one of the four emotions.



text	label
You must be knowing #blithe means (adj.) Happy, cheerful.	joy
Old saying 'A #smile shared is one gained for another day' @YEGlifer @Scott_McKeen	joy
Bridget Jones' Baby was bloody hilarious 😂 #BridgetJonesBaby	joy
@Elaminova sparkling water makes your life sparkly	joy
I'm tired of everybody telling me to chill out and everythings ok. no the fuck its not. I'm tired of...	joy

Figure 05. An example of the WASSA-2017 dataset

5. **GoEmotions Dataset**

Link: https://huggingface.co/datasets/go_emotions

Dataset Storage Size: 28.3 MB

Data Set Size and Split Information: There are 265488 observations with a split of 43410, 5426, and 5427 records for training, validation, and testing respectively.

Description: This dataset uses the last taxonomy that was mentioned before, where there are 27 emotion classifications along with being neutral. The data were extracted from English Reddit comments. The original data has some additional features such as the parent ID, link ID, and the author. However, the information provided here regards the simplified version that includes the text, labels, and ID. The mappings of the emotions can be seen in the link provided.

Example: This is an example of GoEmotions dataset showing five observations. The ID is usually removed since it is not a feature of high importance. The text is fed into the model after being processed and the output will be an array of the possible emotion classifications, where each integer corresponds to an emotion.

text string · *lengths* 2 703	labels sequence	id string · *lengths* 7 7
My favourite food is anything I didn't have to cook myself.	[27]	eebbqej
Now if he does off himself, everyone will think hes having a laugh screwing with people instead of actually dead	[27]	ed00q6i
WHY THE FUCK IS BAYLESS ISOING	[2]	eezlygj
To make her feel threatened	[14]	ed7ypvh
Dirty Southern Wankers	[3]	ed0bdzj
OmG pEyToN iSn'T gOoD eNoUgH tO hElP uS iN tHe PlAyOfFs! Dumbass Broncos fans circa December 2015.	[26]	edvnz26
Yes I heard abt the f bombs! That has to be why. Thanks for your reply:) until then hubby and I will anxiously wait 😬	[15]	ee3b6wu
We need more boards and to create a bit more space for [NAME]. Then we'll be good.	[8, 20]	ef4qmod

Figure 06. An example of the GoEmotions dataset

The Chosen Dataset

We decided to use the CARER dataset, which is the third dataset mentioned in the survey of datasets. As previously mentioned, it contains English Tweets and has six classifications. There are two columns in the dataset, which are “text” and “label”. This dataset was used as it is commonly used and it is feasible. It has good storage size, which is important as resources are limited. Furthermore, it is well-documented and easily accessible. There are three csv files that can be downloaded from the link provided in the datasets’ survey section, which represent the training, validation, and testing data.

Surveys of Models

Paper (A): Using BERT to extract emotions from personal journals [1]

Paper Brief

In this research, the authors developed and trained an emotion classification model, achieving promising results on two publicly available datasets: CARER and SemEval.

Network Architecture Details

Building on SmallBert, the researchers added a dense neural layer with 6 output neurons for classification. Then softmax is applied to interpret network outputs as probabilities. The researchers took advantage of the framework, using TensorFlow cache to enhance performance. The loss function implemented is categorical cross entropy and the optimizer is Adam (Adaptive Moment Estimation). As for regularization, the researchers incorporated a dropout layer which deactivated some neurons during training iterations. Based on experiments, the optimal initial learning rate is 5e-5. The following section shows the model performance.

Model Performance

It is important to mention that the model was trained on two separate datasets rather than being combined.

CARER Dataset:

- The achieved accuray on the test data is **92.2%**.
- **Figure 07** displays the confusion matrix, illustrating the classification of the six emotions.

	anger	fear	joy	love	sadness	surprise
anger	256	10	2	0	7	0
fear	5	202	0	0	7	10
joy	3	1	655	31	3	2
love	0	0	28	130	1	0
sadness	15	6	3	3	554	0
surprise	0	11	4	0	3	48

Figure 07. Confusion matrix showing classification of each class.

- The researchers focus on the model's confusion between Anger & Fear, and the fact that Love & Joy are mixed. The researchers mention that one reason for confusion of emotion classification is that people might see the same sentence differently. Another insightful observation made by the authors is that the number of negated sentences in the training

dataset is not properly represented leaving room for almost completely opposite classification.

- The recall, precision and F1 score of each class is presented in **Figure 08**.

	recall	precision	F1
anger	0.93	0.91	0.91
fear	0.94	0.87	0.90
joy	0.94	0.94	0.94
love	0.81	0.79	0.79
sadness	0.95	0.96	0.95
surprise	0.72	0.80	0.75

Figure 08. Recall, Precision and F1 score of each class.

SemEval Dataset:

- The achieved accuracy is **67.90 %** .

Critical Reflection

This section is dedicated to reflect on the strong points exhibited in the paper and to understand its weak points to be avoided in our contribution to the problem.

- The writers of the proposal appreciate the fact that the two datasets are accessible to the public.
- The model overall performance is competitive and aligns with top results.
- While the code itself is not available, the network architecture is well explained and the used framework (TensorFlow) is mentioned.
- One area of improvement entails including pre-processing details as this makes an impact in the quality of the predictions.
- The weights for the model are not publicly available. As learners, having weights available would have helped us in our first steps following the proposal.

Paper (B): [CARER: Contextualized Affect Representations for Emotion Recognition](#) [3]

Paper Brief

This paper proposed a graph-based algorithm that helped in the preprocessing of text to make contextual representations of the data, where this method is semi-supervised. This paper introduced the CARER dataset, which contains tweets collected from the twitter API.

Preprocessing

As shown in **Figure 09,** there are four stages in the preprocessing stages. The dataset is divided into two subsets: subjective and objective tweets. They are normalized by being tokenized, applying lowercase to all texts, and replacing some unimportant parts such as user mentions and URLs with their corresponding tags. In the graph aggregation, two graphs are created where each graph represents a subset. Aftwards, the two graphs are aggregated to form the emotion graph. It is created so that more subjective content is retained.

This is followed by token categorization where tokens were divided into subject words, like "happy", and connector words such as "the". These were identified using an adjacency matrix and some steps using the eigenvectors. In the last stage, patterns were then formed by providing pattern templates along with using word embeddings for reweighting. Lastly, the tf-idf method was used to reweight patterns, which relies on the importance of patterns for each emotion. Eventually, each pattern has an associated score which determines its importance.

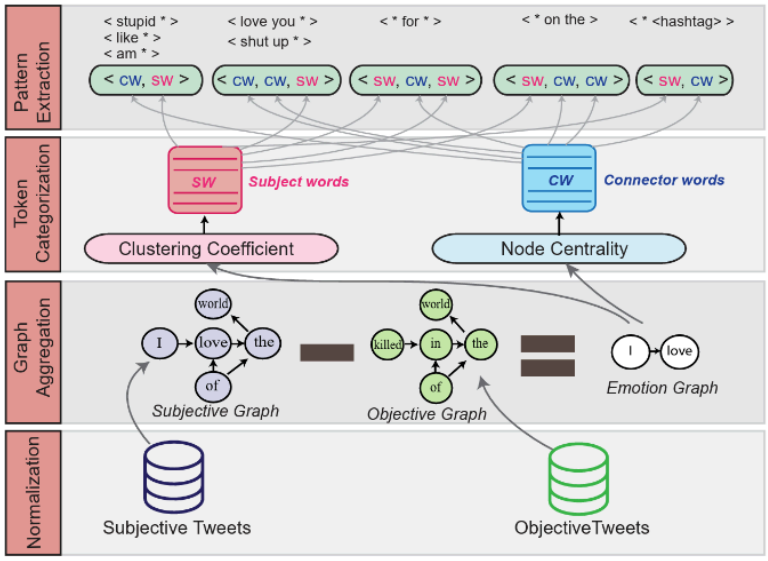


Figure 09. An illustration of the preprocessing stages

Network Architecture Details

The input of the model is a matrix that has the enriched patterns scores. The input matrix is then followed into two 1-D CNN layers that apply filters, where the first has filter size 3 and the second has filter size 16. Afterwards, a ReLU activation function is applied followed by a 1-max pooling layer. The results are then provided to two dropout layers of different dimensions for regularization. The final classifications are provided by using a softmax function.

Training Configurations:

- Activation Function: ReLU
- Optimizer: Adam optimizer
- Number of Epochs: 4
- Regularization: Two 0.5 dropout layers of dimensions 512 and 128 respectively.

Platform Utilized in Implementation:

- Keras was utilized to implement the CNN architecture mentioned above.

Model Performance

For model performance, the proposed one was compared against other deep learning models, such as CNN and another version of the model. The other version fed the input without enriching the patterns. The proposed model had the highest average F1-score of **79%**.

Critical Reflection

This section is dedicated to reflect on the strong points exhibited in the paper and to understand its weak points to be avoided in our contribution to the problem.

- The researchers created a new dataset (CARER) which has been a useful dataset since it has been used in other research papers as well.
- The model overall performance is competitive and is close to the state of the art model with a difference of approximately 0.08.
- The details of the hyperparameters and its values are provided with details in the appendix. However, neither the preprocessing nor the model implementation codes are publicly available making it difficult to reproduce their results.This is crucial especially that the preprocessing steps are complicated and cannot be easily implemented.
- The weights for the model are not publicly available. When checking the associated link that would supposedly have the model implementation and its weights, it is shown that it has the wrong model, with a different architecture for another purpose.

Paper (C): An Optimized Deep Learning Model for Emotion Classification in Tweets [4]

Paper Brief

This paper introduces a hybrid model combining Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN) to address the limitations of each, using sentiment140 dataset. The proposed hybrid technique aims to enhance precision and achieve better results by overcoming the time-consuming nature of LSTM and the inaccurate content representation of CNN.

Pre-processing

Step I: Removing Noise

Given the nature of the real tweets, which includes certain words considered as noise that need to be eliminated, the researchers removed the following during the data preparation process: hashtags, URLs, slangs, acronyms, and other lexical complexities. They relied on a python package named tweet-processor 0.60.

The following steps were conducted:

- Tokenizing words; which mean breaking down text intro units that are called tokens.
- Eliminating hashtags (e.g. #Happy) and URLs (e.g. links to other sites).
- Modifying uppercase to lowercase.
- Removing special words for Twitter (RT which stands for retweet).
- Eliminating unnecessary words and those that have multiple pronunciations.
- Deleting stop words (e.g. a, that, the).

Step II: Test Normalizing

As Twitter users consider it their microblog, many engage in creative expression by writing words with repeated syllables, making it challenging to recognize. Consequently, researchers resorted to lexicon correction. They elaborate on their preprocessing step using **Figure 10** in the following diagram.

Figure 10. An illustration showing the text normalization step

Model Architecture

The dataset is divided into K parts. The model input layer relies on pre-trained word embeddings obtained from Google News, Word2Vec. Each part is passed on its own to CNN which captures the semantic meaning and relationships of the words. Next, a max pooling layer is applied to produce a vector of smaller dimensionality. Afterward, a sequential model is implemented. For this sequential part, the first subset is provided to the LSTM model. The second model receives the output of the 1st LSTM model along with the output of the max pooling layer for the 2nd part. These two parts of data are provided as input to the second LSTM model. This follows for all LSTM models. Afterwards, the output of the last LSTM model is fed into the linear decoder. The model architecture is demonstrated in **Figure 11**.

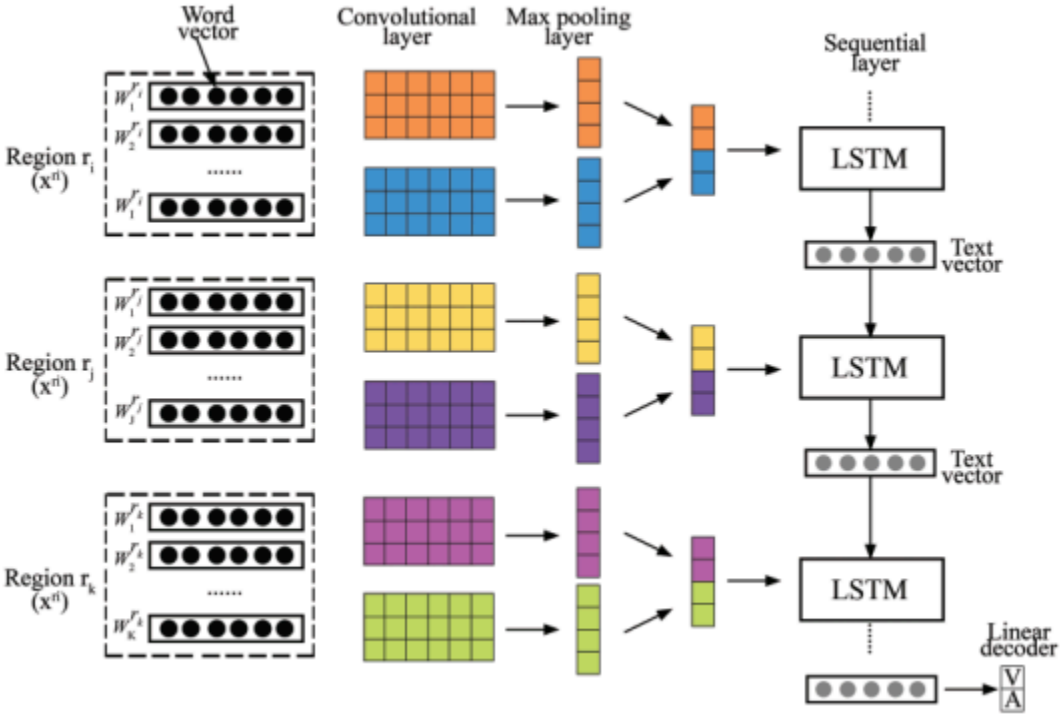


Figure 11. An illustration of the architecture

Model Performance

The following metrics were used to evaluate the model: Accuracy (**85%**), Precision (**90%**), Sensitivity (**85%**), F1 Score (**87%**).

Critical Reflection

This section is dedicated to reflect on the strong points exhibited in the paper and to understand its weak points to be avoided in our contribution to the problem.

- We appreciate that the authors of this paper included the pre-processing steps.
- The used dataset is available on Kaggle and easily accessible.
- The visualization of the model architecture of the model helps in understanding key concepts.
- The model architecture is creative and opens our eyes into hybrid techniques.
- Some of the drawbacks include unavailable code and weights for public access.

Paper (D): GoEmotions: A Dataset of Fine-Grained Emotions [1]

Paper Brief

The researchers introduced a new dataset called GoEmotions, which is characterized by having much more emotion classifications than just the six basic ones. To be more specific, the dataset has 27 different emotions along with the neutral one. These emotions are used to label 58000 English Reddit comments. The dataset was then fed into a BERT model architecture to classify the Reddit comments.

Pre-Processing

1. *Filtering Data*
   Reddit is a platform that is known to have an offensive language in several of its posts and comments. These were filtered out using predefined lists of offensive, religious and racist terms. Nonetheless, some vulgar comments remained in the dataset to represent the negative emotions as well making the dataset more reliable.
2. *Sentiment and Emotion Balancing*
   In this phase, subreddits that were rarely represented in the dataset, whether positive, negative, ambiguous, or neutral, were eliminated to avoid over-representing noise which might fail our model to perform its intended task. Afterwards, each comment was assigned one or two emotion labels.
3. *Subreddit Balancing and Masking*
   In the final phase, more filtering took place in an attempt to reduce the imbalance of the data. User names and religions were also masked with specific tokens using a BERT-based Named Entity Tagger. Also, all neutral comments were removed as they were not the main concern of the classification problem.

In addition to the original dataset, two other datasets were created using different taxonomies, where one used Ekman's taxonomy with the six emotions and the other labeled comments using the sentiment level, where there are four categories: positive, negative, ambiguous, and neutral. The three datasets were used for the model to evaluate the different classifications on the model.

Model Architecture

After preprocessing the data, it was fed into a pre-trained BERT model, followed by a dense output layer for fine tuning. Afterwards, a sigmoid cross entropy loss function was applied to allow multi classification. This proposed model was compared to a bidirectional LSTM, which was also trained on the three groups of data.

Training Configurations:

- Loss Function: Sigmoid cross entropy loss function
- Number of Epochs: 4
- Learning Rate: 5e-5

Platform Utilized in Implementation:

- There are several implementations of this architecture using different platforms. However, the adopted platform that we will use in the project utilizes PyTorch.

Model Performance

1. *Original Data*
   When providing the original data to the model described above, it produced a macro average F1-score of 46%. Furthermore, its precision and recall are 40% and 63% respectively.
2. *Ekman's Taxonomy Data*
   When providing Ekman's taxonomy data to the model described above, it had a macro average F1-score of 64%. As for the precision and recall, they were 59% and 69% respectively.
3. *Sentiment Based Data*
   As for the sentiment-based data, the model had a macro average F1-score of 69%. Furthermore, it resulted in a precision and recall scores of 65% and 74%.

As expected, the sentiment-based version had the highest performance metrics since it only had four labels unlike the other two groups of data.

Critical Reflection

This section is dedicated to reflect on the strong points exhibited in the paper and to understand its weak points to be avoided in our contribution to the problem.

- The researchers created a new dataset (GoEmotions) which has a variety of emotions, which made it more realistic to the real human's emotions.
- While the model's performance was not the best, this has left room for creativity and manipulating the model, which can be performed in our project.
- The three versions of data along with the code are publicly available, easing its reproducibility.

- The weights are not publicly available.
- Lack of visualizations in the preprocessing and model architecture. These figures often help in understanding the paper. However, their absence resulted in difficulty in comprehension and thus, giving a harder time in modifying the code. Furthermore, while the code is publicly available, it is not well-documented making other researchers having difficulty in keeping up with the code.

Paper (E):Emotion Analysis From Turkish Tweets Using Deep Neural Networks [7]

Paper Brief

The researchers conducted a study to assess the performance of deep neural networks in analyzing emotions from Turkish tweets. They investigated three different architectures: artificial neural network (ANN), convolutional neural network (CNN), and recurrent neural network (RNN) with long short-term memory (LSTM). The two used datasets are TREMO and TURTED.

Model Architecture

Neural Network I: ANN

Figure 12 shows the model architecture, which consists of three layers: input, hidden, and output. The input layer contains 1000 neurons, followed by a hidden layer with 128 neurons, and finally an output layer with 6 neurons. A dropout rate of 20% is applied to prevent overfitting. ReLU and SoftMax activation functions are used for the hidden and output layers, respectively. Additionally, Adamax is employed for optimization, and the Categorical Cross-Entropy function serves as the loss function.

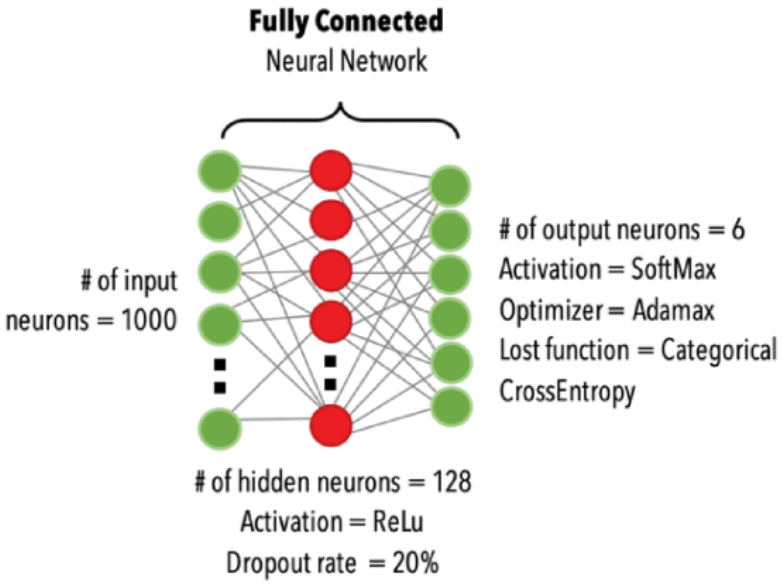


Figure 12. Network Architecture

Neural Network II: CNN

As illustrated in **figure 13**, the process begins with the creation of an embedding layer using a vector space of 8 dimensions and an input sequence length of 100. Following this, a convolutional layer is defined with parameters including 32 filters, a ReLu activation function, and a kernel size of 8. A pooling layer is introduced, and its output is then flattened, which will be fed into a fully connected network. The architecture of this network is explained above

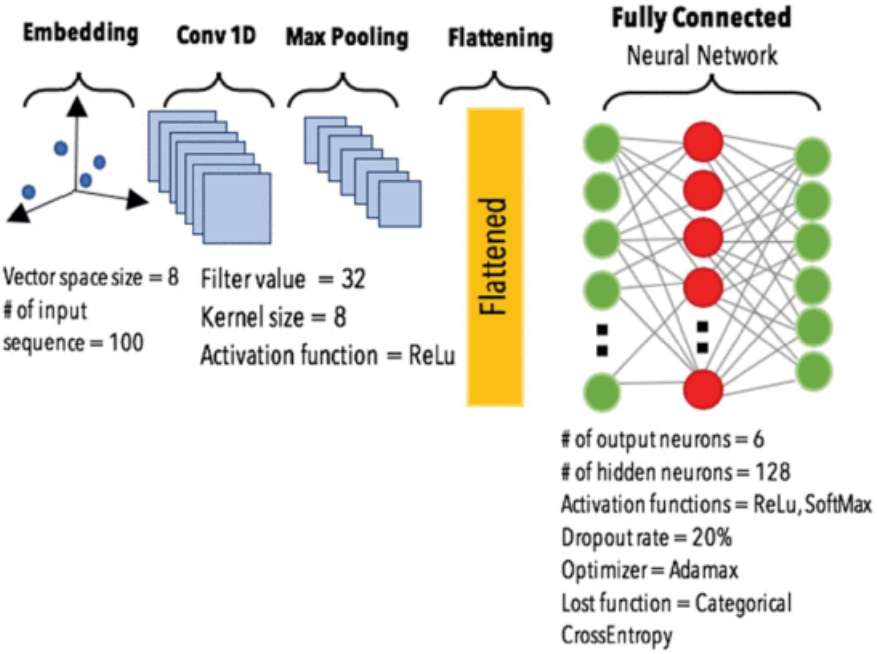


Figure 13. Model Architecture

Neural Network III:RNN

The model architecture is demonstrated in **Figure 14**, similar to CNN an embedding process takes place with the same details. Results of this process are passed to a Long Short Time Memory (form of RNN) which has 128 neurons. Then, a dense neural network is introduced, with the architecture previously detailed.

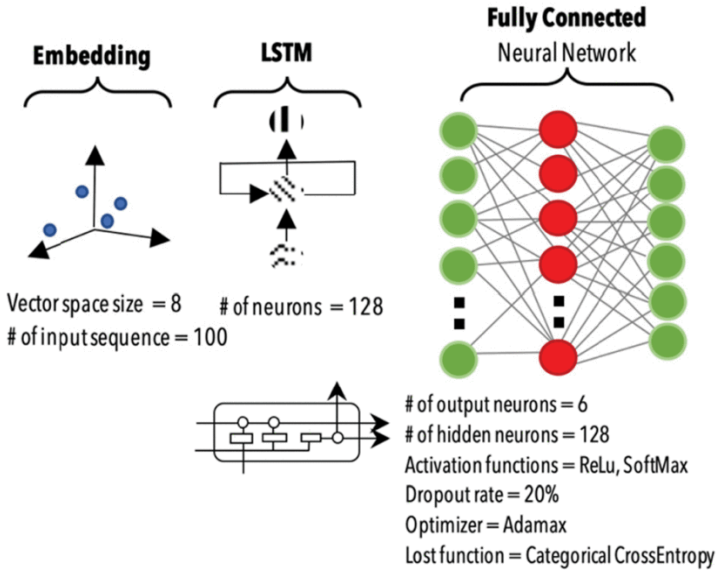
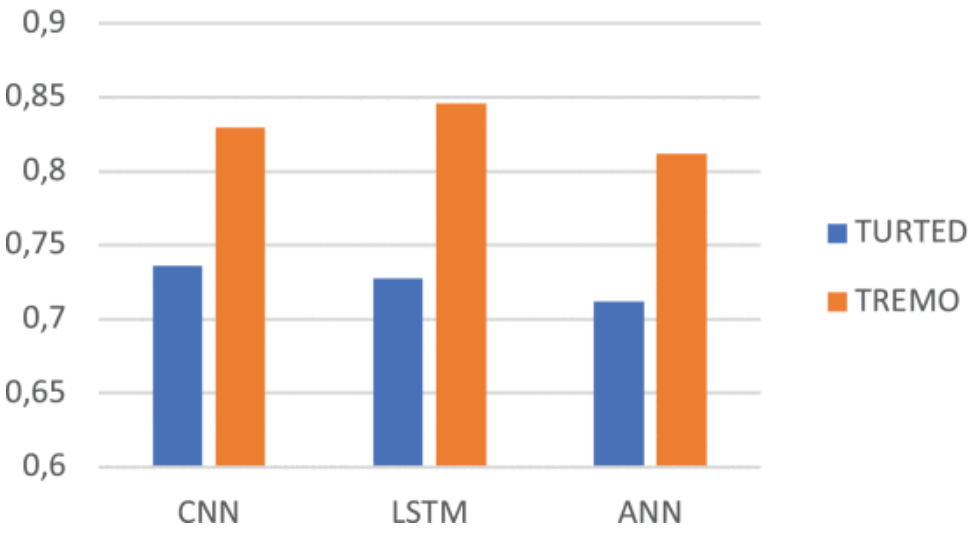


Figure 14. Model Architecture of RNN

Model Performance

The model has been tested on two separate datasets: TURTED and TREMO. It is important to note that all performance measures are micro-averaged values.

Figure 15 includes the accuracy achieved by each model. To explain, CNN performs the best for TURTED, while LSTM has the highest accuracy for TREMO



Fugue 15. Accuracy of each dataset using CNN, LSTM, ANN.

Critical Reflection

This section is dedicated to reflect on the strong points exhibited in the paper and to understand its weak points to be avoided in our contribution to the problem.

- The paper is well written and gives a detailed explanation of the neural networks. The given visualization plots are great aspects of the paper.
- While the paper provides links or references to the two datasets, accessing them is challenging.
- The code and the weights are not publicly available which is a setback.

Summary of Various Papers

Paper	Datasets	Performance
Paper (A)	CARER	Accuracy: 92.2%
	SemEval	Accuracy: 67.90%
Paper (B)	CARER	F1 Score: 79%
Paper (C)	Sentiment140	Accuracy: 85% Precision: 90% Sensitivity: 85% F1 Score: 87%
Paper (D)	GoEmotions	Precision: 40% Recall: 63% F1-score : 46%
	Ekman's Taxonomy Data	Precision: 59% Recall: 69% F1-score: 64%
	Sentiment Based Data	Precision: 65% Recall: 74% F1-score: 69%
Paper (E)	TURTED	CNN Accuracy (≈ 74%) LSTM Accuracy (≈ 73%) ANN Accuracy (≈ 72%)

	TREMO	CNN Accuracy (≈ 83%) LSTM Accuracy (≈ 85%) ANN Accuracy (≈ 71%)

The Chosen Model

We decided to choose the model architecture implemented in paper D, which utilizes the BERT model along with other models for fine tuning and classification [1]. Since there are several versions of the code as it was implemented on various platforms, we decided to use the code implemented using PyTorch. Since the dataset that we will use, which is the CARER dataset, is not the same as the one in paper D, our first contribution will be integrating the code with our dataset and ensuring that the preprocessing, training, and testing are running smoothly without errors.

The main reason for choosing this model is its feasibility; having the code publicly available has significantly influenced the choice of deciding our model as many other papers suffered from having their codes unavailable. Fortunately, the code of the BERT-based architecture can be easily accessed and downloaded from GitHub. Another reason to choose this architecture is that it provides room for us to add more layers, fine tune the hyperparameters, and manipulate the data in an attempt to achieve higher results.

The only potential problem faced when using this model is that the weights are not provided. However, upon discussion, it was decided that it is not a problem since we will use a different dataset. Consequently, we will need to train the model from the start, and the weights of the model can then be documented using the CARER dataset.

Proposed Contributions

In this section, we outline the initiatives planned for our project. Our objective is to enhance our skills through understanding and practical experimentation. Before implementing any of the following initiatives, we will integrate the CARER dataset to our cohen model, which is the BERT-based model. This step is crucial to make sure that the code can work for the CARER dataset, and not limited to the GoEMotions dataset. Furthermore, the resulting model will be our baseline against which all models created from our initiatives will be compared against.

Initiative I: Modifying Layers

Below are examples of the models that we will be experimenting with. Furthermore, we might need to add layers between each, such as a dropout layer for regularization. This will be further explored in the future.

- Inspired by Papers (D) and (E), we intend to conduct three experiments, each involving BERT paired with one of the mentioned neural networks. Specifically, we will use our chosen dataset CARER with BERT and a CNN layer in one experiment. In another, we will explore the combination of BERT with RNN, specifically LSTM. The third experiment will involve BERT paired with ANN.

- We will also experiment with BERT, incorporating CNN followed by LSTM, inspired by the approach outlined in Papers (D) and (C).

- Using the pre-processing and word embedding techniques used in Paper (C), we will try building a model with CNN and LSTM.

Initiative II: Fine Tuning Hyperparameters

- We wish to experiment with parameter fine-tuning, such as learning rates, batch sizes, number of epochs, activation functions, etc.

Initiative III: Pre-processing

- We aim to compare the impact of pre-processing on our chosen dataset, CARER. We will use the steps from Paper (C) on the models implemented in Paper (D). We will evaluate the results to better understand which processing is better.

- We wish to experiment stemming, where words like 'running' or 'ran' become 'run'. The aim is to detect whether reducing the various tokens through the method stemming would affect the model's performance.

Initiative IV: Generalization

- We will attempt to apply our best-performing model to another dataset. Initially, we propose using the ISEAR dataset.

Initiative V: Reporting

- Our dedication to solving the problem does not stop at experimenting with code; we aim to document our code thoroughly and make the weights and code available for future problem solvers.

Chosen Evaluation Metrics

As for evaluating our model, our primary evaluation metric is the F1-score since it is highly effective with datasets that have various distributions of its labels. However, to make the results comparable to most of the papers, we will also include all of the metrics mentioned above to be able to compare our results with other researchers. Therefore, we will be using precision, recall, and accuracy. Moreover, we will be using the confusion matrix in order to easily visualize the TP,NP,TN, and FN. While the evaluation metrics are values, we will be developing bar graphs to enable comparing the different models that we will develop throughout the project.

Graduation Project

Graduation Project Title: Prediction of Epileptic Seizures Using Machine Learning

Project Dataset: CHB-MIT Scalp EEG Database

Project Main Idea: Epilepsy affects an estimated 50 million individuals worldwide, significantly impacting their quality of life. This neurological disorder is characterized by recurrent seizures varying in severity, from mild loss of awareness to life-threatening status epilepticus, a prolonged seizure that requires emergency medical attention due to its significant risk on the patient's life. To address this challenge, our research focuses on developing predictive machine learning models that allow detection before seizure onset to mitigate the seizure's potential devastating consequences. By providing accurate classifications, our approach aims to enhance patient care and quality of life. Our work contributes to the ongoing effort to improve epilepsy management and empower patients with better treatment options.

Link to Literature Review: Literature Review

Graduation Project Teammates: Habeeba Mohamed & Mona Ahmad

Graduation Project Supervisor: Dr. Seif Eldawlatly

Previous Projects

The two researchers involved in this emotion classification project have not collaborated on similar machine learning projects in the past. Nonetheless, we will highlight some of our previous projects to familiarize the reader with our skills and expertise.

- Machine learning Project by Habeeba Mohamed & other team members.

  Title: Machine Learning Application in Predicting Used Cars Price

  Project Dataset: US Used cars dataset

  Project Main Idea: The project focuses on forecasting the prices of used automobiles in the United States by considering factors such as fuel economy, engine type, and power horse. The aim is to provide consumers with valuable information to make informed decisions in the realm of used-car transactions. Motivated by the lack of extensive research on used car prices, the study seeks to contribute to the understanding of this domain, exploring machine learning as a problem-solving method. The primary objective is to assess the effectiveness of various machine learning approaches in predicting used car prices using a dataset of US used cars, offering insights into their performance and guiding future applications in similar challenges.

  Project Supervisor: Dr. Hossam Sharara

- Machine learning Project by Mona Ahmad & other team members.
  Title: Machine Learning Models on Titanic Data
  Project Dataset: Titanic Dataset
  Project Main Idea: The project aims to predict whether a passenger survived the sinking of the Titanic or not based on the remaining features. This project is mainly assessing the students to test various models studied during the course as well as testing the main data analysis phases. First, EDA and data cleaning were performed to truly understand the data, and manipulate it so that it is ready to be the input of the models. Afterwards, several machine learning and deep learning models were developed in an attempt to compare between their performances. Some examples of machine learning and deep learning models include linear regression and artificial neural network (ANN). Hyperparameters were also manipulated in order to show their effectiveness in improving or reducing the performance of the neural network.

  Project Supervisor: Ahmad Shawky Moussa

Contributions

Below are the contributions of each team member regarding the initial proposal:

1. *Team Member Name: Mona Ahmad Kamel*
    - Problem statement,motivation, and input/output examples.
    - Survey of evaluation metrics.
    - Current state-of-the-art results.
    - Survey of the available datasets and our chosen dataset.
    - Paper (B) and (D) in the survey of models.
    - Chosen model and evaluation metrics.
    - My machine learning project in the “previous projects” section.
2. *Team Member Name: Habeeba Hossam Mohamed*
    - Paper (A), (C), (E) in the survey of models.
    - Summary of Various Papers
    - Graduation Project
    - Proposed Contributions
    - My machine learning project in the “previous projects” section.

References

1. Demszky, D., Movshovitz-Attias, D., Ko, J., Cowen, A., Nemade, G., & Ravi, S. (2020). GoEmotions: A dataset of fine-grained emotions. *arXiv preprint arXiv:2005.00547*.
2. Razek, M. A., & Frasson, C. (2017). Text-based intelligent learning emotion system. *Journal of Intelligent Learning Systems and Applications*, *9*(01), 17-20.
3. Saravia, E., Liu, H. C. T., Huang, Y. H., Wu, J., & Chen, Y. S. (2018). Carer: Contextualized affect representations for emotion recognition. In *Proceedings of the 2018 conference on empirical methods in natural language processing* (pp. 3687-3697).
4. Singla, C., Al-Wesabi, F. N., Pathania, Y. S., Alfurhood, B. S., Hilal, A. M., Rizwanullah, M., ... & Mahzari, M. (2022). An Optimized Deep Learning Model for Emotion Classification in Tweets. *Computers, Materials & Continua*, *70*(3).
5. Ovidiu-Mihai, V., Tudor-Alexandru, I., & Petrescu, M. (2022, September). Using BERT to extract emotions from personal journals. In *2022 IEEE 18th International Conference on Intelligent Computer Communication and Processing (ICCP)* (pp. 89-94). IEEE.
6. Singla, C., Al-Wesabi, F. N., Pathania, Y. S., Alfurhood, B. S., Hilal, A. M., Rizwanullah, M., ... & Mahzari, M. (2022). An Optimized Deep Learning Model for Emotion Classification in Tweets. *Computers, Materials & Continua*, *70*(3).
7. Tocoglu, M. A., Ozturkmenoglu, O., & Alpkocak, A. (2019). Emotion analysis from Turkish tweets using deep neural networks. *IEEE Access*, *7*, 183061-183069.