May 2023

Emotion Classification and Intensity Prediction on Tweets

Sharath Chander Pugazhenthi
University of Wisconsin-Milwaukee

EMOTION CLASSIFICATION AND INTENSITY PREDICTION ON TWEETS

by

Sharath Chander Pugazhenthi

A Thesis Submitted in

Partial Fulfillment of the

Requirements for the Degree of

Master of Science

in Computer Science

at

The University of Wisconsin–Milwaukee

May 2023

ABSTRACT

EMOTION CLASSIFICATION AND INTENSITY PREDICTION ON TWEETS

by

Sharath Chander Pugazhenthi

The University of Wisconsin–Milwaukee, 2023
Under the Supervision of Professor Rohit J. Kate

The task of finding an emotion associated with the text from individuals on a social media platform has become very crucial as it influences the current state of mind of a particular individual in real life. It also helps one to understand social behavior at a given point in time. Microblogging platforms like Twitter serves as a powerful tool for expressing one's thoughts. Several work have been done in classifying the emotion associated with it. The thesis comprises of a system that first classifies the tweet into one of the four emotions - anger, joy, sadness, and fear with good accuracy. It is also important to understand the intensity of the emotion in determining how strong one's tweet is. Hence, the second phase of the system is built using regressors that help in predicting the intensity of the emotion in the tweet. Both the classification and intensity prediction systems were evaluated on a competition dataset and the regressors outperformed the best system from the competition.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

NLP	Natural Language Processing
BERT	Bidirectional Encoder Representations from Transformers
WASSA	Workshop on Computational Approaches to Subjectivity, Sentiment, and Social Media Analysis
TBED	Text-Based Emotion Detection
BWS	Best-Worst Scaling
MSE	Mean Squared Error
SHR	Split-Half Reliability

ACKNOWLEDGMENTS

I would like to sincerely thank my advisor Prof. Rohit J. Kate, whose constant support and mentorship helped me successfully complete my thesis. His tireless effort in guiding me on the right path helped me learn the process of research at its best quality. I appreciate all of his efforts in enhancing my thoughts throughout the process.

I would also like to extend my gratitude to all the committee members of my thesis: Prof. Susan Mcroy and Prof. Tian Zhao for their valuable thoughts, suggestions, and support.

I also thank the College of Engineering and Applied Sciences and the Centre for International Education at the University of Wisconsin-Milwaukee for providing their support throughout my graduate studies and building my career.

Finally, I thank my parents, Mr. Pugazhenthi Punniyaseelan & Mrs. PT Chithra, my brother, Vishal, my paternal grandparents, Mr. Punniyaseelan & Mrs. Nagamal and maternal grandparents, Mr. Thangavel Raj & Mrs. Puspa for all their constant support, and motivation.

Introduction

1.1. Background and Problem Statement

In Natural Language Processing, sentiment analysis is used to detect a sentiment level associated with text data whether it is positive, negative, or neutral. While going deep into this, detecting emotions is found to be more useful [2] as it provides an exact insight into the individual's feelings. Twitter is one of the most used social or microblogging platforms where users express their opinions based on their personal or the current state of society. This could be influenced based on any political changes, sports, disasters, and health. These opinions are associated with emotions that find applications in various business that runs on user satisfaction as in online commerce, understanding social welfare, and public health.

Tweets are most often short text messages which carry a state of emotion with them. One can view Twitter as representing people in groups like a set of people belonging to certain countries, communities, and regions of the world. Hence These tweets are rich with information representing a mass group like their emotions, state of mind, moods, etc. [3]. In this thesis, I'm building a system for automatically classifying the emotion and predicting the intensity associated with the tweets. Systems built on this principle could be of potential use in applications like e-commerce, well-being counseling apps, mental health apps, etc. To elaborate on this, if an e-commerce website were able to find the emotion of the people on their brand, it helps them drive business in the right direction. But, when they could also see the intensity associated with the emotion, it will help them build or improve their business model in the earliest possible time. The mood of society could be detected from the tweets

helping scientists or policymakers build better living conditions or quality of life for the public [3]. The intensity is measured in a numerical value from 0 to 1 where 0 is less intense and the intensity increases as it moves up to 1.

The intensity of an emotion conveys the amount or degree of the emotion expressed like very sad, slightly angry, absolutely happy, etc. Tweets often associated with emojis, and hashtags boost one's emotional expressiveness [1]. Hence, a robust system for automatically predicting the intensity of emotions helps in further understanding the text data. An example tweet for each emotion along with their intensity is shown in Table 1.

Table 1: Sample tweets for sadness, anger, joy, and fear from the WASSA-2017 [12] dataset.

Text	**Emotion**	**Intensity**
Panic attacks are the worst. Feeling really sick and still shaking. I should be a sleep. #anxiety #depression	Sadness	0.917
I hate my lawn mower. If it had a soul, I'd condemn it to the fiery pits of Hell.	Anger	0.833
I feel so blessed to work with the family that I nanny for ♡ nothing but love & appreciation, makes me smile.	Joy	0.938
I have another test tonight #nervous	Fear	0.812

Table 2 shows a few examples of tweets with anger emotions of varying intensity levels.

Table 2: Sample tweets for anger with variations in intensity from the WASSA-2017 [12] dataset.

Text	**Emotion**	**Intensity**
wanna go home and focus up on this game . Don't wanna rage at all	Anger	0.375
The war is right outside your door #rage #USAToday	Anger	0.500
@easyJet Hi folks. Flight is going to be over an hour late departing from INV (EZY864), how do we go about getting a refund please?	Anger	0.625
Dear hipster behind me at the game I am finding it very hard to pay attention while you talk about the politics of grapes of wrath. SHUT UP!	Anger	0.854

Hence, experiments were conducted to build an emotion classifier with good accuracy which is used for detecting the type of emotion from the tweets. A state-of-the-art intensity predicting regressor model is built to convey the severity of emotion felt in the tweet. Finally, these two modules have been combined as a single system and their results are analyzed.

1.2. Literature Review

This section discusses the related works of emotion detection and intensity prediction. However, these tasks have been proposed by many authors separately but not as a combination.

An emotion prediction model was built in [4] by harnessing the Twitter dataset. They have built a very large Twitter dataset labeled with emotions and employed deep-learning models like Gated Recurrent Neural Networks (GRNN) to detect emotions. Their work was based on the eight emotions in the psychological theory of emotion. In [5], the authors have proposed a multi-class emotion detection system using Twitter messages or tweets by cross validating the models trained on different labeling for the fundamental emotions via a method called distant supervision [6]. In a similar vein, Bollen et al explored the mood patterns of the public which is evidenced by the emotion analysis using their Twitter dataset. In [8], a large emotion dataset was automatically created by leveraging the hashtags that reacted to emotions available in the tweets. Two machine learning algorithms: Multinomial Naïve Bayes(MNB) and LIBLINEAR[8] were used to detect emotions from the Twitter dataset built by them. An empirical study on a supervised machine learning approach to classify emotions for text data has been proposed in [9]. They discuss the problems associated with it by having the goal to classify the emotional affinity of sentences in children's fairy tales. In [3], a comparison of the performance of different machine learning algorithms with their system on emotion classification using the Twitter dataset has been done. EMOTEX is the model proposed by them where they have used the Circumplex model to define the emotional states of humans while building their dataset. Automatic emotion analysis of news headlines has been built in [15] where the data has been gathered and annotated from sources like New York Times, BBC News, and Google News Search Engine.

Thelwell et al [16] developed an algorithm called SentiStrength to identify the strength of sentiments from informal English messages. This algorithm was exclusively focused on user behaviors rather than commercially oriented. It was then applied to MySpace comments

where positive emotions were predicted with 60.6% accuracy and negative emotions with 72.8% accuracy. Works on emotion intensity prediction are not very common. However, Mohammad et. al[10], developed the first emotion intensity dataset of tweets using best-worst scaling (BWS). This method was used for obtaining a fine-grained intensity score for each emotion annotated tweet. They have concluded by determining the most useful features required for emotion intensity detection tasks. The dataset developed in this paper was utilized for a competition for building a regressor for detecting emotion intensity. Out of the 22 teams that participated in the competition, the best Pearson correlation coefficient obtained by the winning team was 0.747 [1]. The data developed by Mohammad et al have been partitioned into training, development, and test sets for the sake of the competition. The same data is used in this thesis for training and evaluation for the task of emotion intensity prediction.

1.3. Motivations and Objectives

Emotion detection plays a major role in analyzing social public behavior when it comes to social media such as Twitter. With powerful tools to classify emotions, one can gain insight into the behavior of users through their tweets. Emotion detection also finds applications in brand management as it helps the company address dissatisfied customers with top priority. Apart from this, the lack of systems that perform emotion intensity prediction of those tweets is a major motivation behind this research work. Given the intensity and emotion of customer utterances, immediate action can be taken by the brands when it comes to resolving problems with dissatisfied customers. Analyzing the emotion and intensity among

tweets helps understand the user's mind like what kind of action could trigger their emotions.

1.4. The Main Contributions

The main goal of the thesis is to build a system that predicts emotion intensity of a sentence. The WASSA-2017 competition dataset was used for this purpose. The competition focuses on building a system that can be used to predict the intensity of emotion felt in tweet sentences given the tweet and emotion. The approaches and techniques used in this thesis provide better results than the top systems submitted in the competition. This thesis also demonstrates an approach for building a classifier to classify the type of emotion with higher accuracy. This work also introduces a way of predicting emotion or its intensity by giving a second sentence with an emotion as part of the input to a transformer-based system. Finally, an experiment was conducted by combining the classifier model along with the regressors for each emotion as a single system.

Materials and Methods

2.1. Dataset

The dataset used in the thesis is from a competition titled WASSA-2017 Shared Task on Emotion Intensity. Most of the datasets present today are annotated categorically with labels conveying only the type of emotion present in a text. Hence, they are viewed as a classification problem. However, the authors of [1] have developed a dataset with an extra feature, the intensity of emotion. This is very important in various practical applications. Hence, the main goal of the competition was to develop a system that automatically predicts the intensity of emotion in a tweet given the tweet and the emotion. The authors of the competition had developed datasets for training, validating, and testing. All three datasets had 4 types of emotions in them, namely, anger, joy, sadness, and fear along with their intensity values ranging from 0 to 1. The validation dataset had very few examples when compared to the training dataset and hence when performing classification and regression tasks both the training data and the validation data were combined and then they were split with a validation size of 0.2. The dataset is publicly available in [12]. The description of the different datasets is shown in Table 3.

Table 3: Count of tweets for each emotion in the train, test, and validation set.

	Anger	**Joy**	**Fear**	**Sadness**
Training Data	857	823	1147	786
Validation Data	84	79	110	74
Testing Data	760	714	995	673

Best-Worst Scaling is an algorithm used in the NLP community for annotating words like word sense disambiguation [17], word sentiment intensity [18], etc. This method was used by the authors of [10] to annotate the WASSA-2017 dataset with the intensity of the emotion. The reliability of these annotations was assessed by calculating the average of *split-half reliability* which is a commonly used approach to find consistency [1].

2.2. Definitions

Following are the definitions of important terms used in this work.

2.2.1. Classifier

The task of assigning a class or category to data points or observations is called classification. An algorithm that automates classification by assigning data to one or more categories is called a classifier.

2.2.2. Regression Analysis

A statistical method that helps build relationships between a numerical target and the dependent variables is called regression analysis. It can be used to model the future relationship between the variables and for assessing the strength between them.

2.2.3. BERT

Bidirectional Encoder Representations from Transformers (BERT) [14] is a language representational model. Its framework has two steps – pre-training and fine-tuning. First it is pretrained with large amount of unlabeled text from which it learns a general language model. This pre-trained BERT with an addition of one output layer can be finetuned according to a specific task. It has obtained excellent results on several NLP tasks. The use of

pre-trained representations reduces complicated engineered task-oriented architectures becoming the first fine-tuning-based representation model [13]. For pretraining the BERT-Base model is used. BERT-base is trained on a book corpus of nearly 800 million words. This makes BERT very strong in natural language understanding.

2.2.4. Confusion matrix

The performance of a supervised learning algorithm can be visualized in a specific table format called a confusion matrix. It is an N X N matrix where N represents the total number of output classes. The rows represent the actual class, and the columns represent the predicted class. An example of a confusion matrix for binary classification is shown in Figure 1.

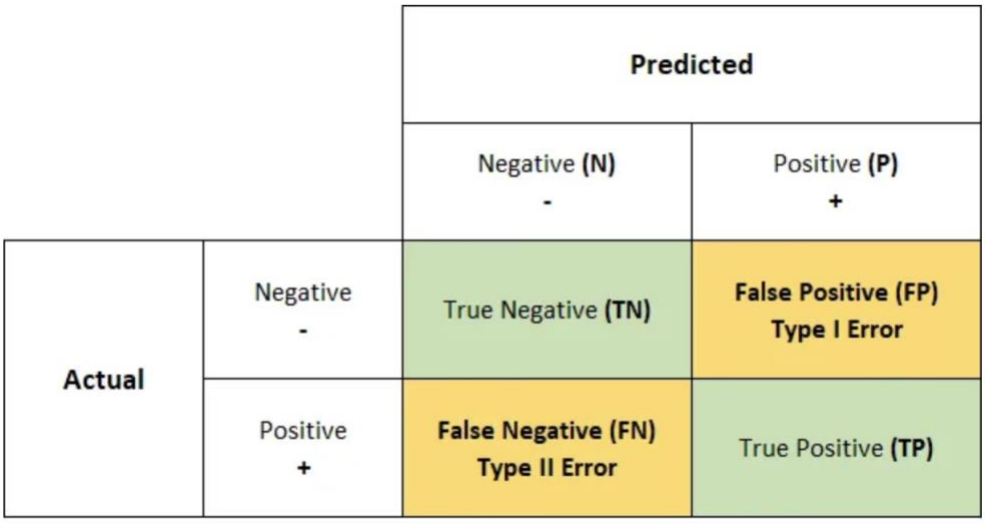

Figure 1: Confusion matrix for Binary Classification problem (2X2 matrix).

A classifier is said to have a good performance if it has high True Positive and True Negative rates [19].

2.2.5. Precision

Precision is one of the performance metrics which tells the proportion of the correct instances from positively identified instances [11].

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

2.2.6. Recall

Recall helps to answer the proportion of actual positives that were identified correctly [11].

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

2.2.7. Mean Squared Error

In regression, Mean Squared Error is a metric that is used to calculate the closeness between the set of points and the regression line. The difference between the distance of the point to the regression line is the error. This error is squared to avoid negative values. The average among the set of errors becomes the Mean Squared Error. Generally speaking, the lower the MSE the better the performance of the regressor.

2.2.8. Pearson Correlation Coefficient

This metric is measured between two variables and is used to determine the linear correlation between them. The scores range from -1 to 1 where -1 implies inverse correlation, 1 represents perfect correlation, and 0 means there is no correlation [1].

2.3. Methodology

This section discusses the various techniques applied to the data-cleaning process and techniques employed for the development of classification and regressor models.

2.3.1. Data Preparation

As shown in Table 3 due to the small size of the validation dataset, both the training dataset and the validation set have been combined and then split into train and validation of size 0.2 for classification and regression. The combined total of tweets for training for each emotion is shown in Table 4.

Table 4. Merging training, and validation dataset

Emotion	**Count**
Fear	1257
Anger	941
Joy	902
Sadness	860

Twitter is a collection of random tweets which does not have a common structure. One can freely type in anything which has components other than text like tags, emojis, emoticons, hashtags, usernames, URLs, and informal slang. To train a model, a certain level of preprocessing is required. Some of the rules that I have applied for preprocessing the tweets are:

- The data is first cleaned by removing tags, unnecessary spaces, account tags, and emojis.
- Some tweets tend to have repeated punctuation marks. For example, consider the sentence, "*How are you?????".* These are common in natural language blocks. Hence, these repeated punctuations were removed and replaced by a single punctuation.

- Following this, the Ekphrasis tool was used to further preprocess the sentence. I have employed this tool to remove URLs, email, percent, money, phone numbers, time, and dates.
- The tool was also used to unpack contractions like converting "Can't" to "can not" and shrinking elongated words like "*happyyyyyyyy*" to "*happy*". The tool makes use of SocialTokenizer to understand and tokenize these tweet sentences into their corresponding words which are later combined to get a clean sentence ready for the model to accept.
- Proper whitespaces were introduced in the tweets between continuous hashtags. For instance, hashtags like *"#halloween#festival*" is converted to *"#halloween #festival".* One of the major significant factors in determining the emotion and its intensity in a tweet is the hashtags. Hence, during the preprocessing stage, they are not removed.

2.3.2. Input Formation

The dataset consists of the following features: tweet id, text, emotion, and intensity. Here, tweet id is irrelevant for the models and has been excluded when training the classifier and regressors. When training the classifier, the text sentence/tweet is the input to the model and the *"emotion"* feature is the target. However, when training the regressors, the text sentence and the emotion have been combined to form the input for the model. For instance, consider the sentence "*my mind is raging and I just want to end it all*" and the emotion "*anger*". These are now combined to form "*my mind is raging and I just want to end it all. I am angry"* which now serves as the input to the regressor model. A [SEP] token is introduced between the tweet and the generated sentence from the emotion to distinguish that these are separate

entities and are passed as input to the regressor. The *"intensity"* feature is the target for the regressor models.

2.3.3. Classification and Regressor Model

Language modeling plays a key role in understanding the natural language of machine learning models. The second training step is fine-tuning the BERT model according to a natural language task. In the task of predicting the intensity, I have modified BERT to work as a regressor. One regressor was built for each of the emotions in the dataset - anger, sadness, joy, and fear. For the classification of emotion task, BERT was fine-tuned to work as a classifier to predict the type of emotion with high accuracy.

2.3.4. Experimental Setup

This section discusses the experimental setup for both regression and classification tasks.

2.3.4.1. Regressor

When training the regressors, the text sentence and the emotion have been combined to form the input for the model. These inputs are converted to embeddings of *input_ids* and *attention_masks* for each tweet which is sent as input to the BERT regressor. The *attention_masks* and *input_ids* are generated by making use of the "*Autotokenizer*" from the *Transformers* package of *Huggingface* to prepare the inputs for the model. I have used the pre-trained "*Bert-base-uncased*" model for the Autotokenizer to generate the inputs for the model. The input features of the tokenizer function are shown in Table 5. The output feature is the "intensity" whose values range from 0 to 1.

Table 5: Parameters for the tokenizer which generates inputs for BERT regressor.

Parameter	**Description**	**Value**
add_special_tokens	Whether or not to encode the sequences with the special tokens relative to their model like [CLS], and [SEP]	True
max_length	Controls the maximum length to use by one of the truncation/paddings[SCP1] parameters.	max of all sentence
truncation	Truncate to a maximum length specified with the argument max_length or to the maximum acceptable input length for the model if that argument is not provided	True
padding	Pad to the longest sequence in the batch (or no padding if only a single sequence if provided).	True
return_tensors	If set, will return tensors instead of list of python integers. Returns TensorFlow tf.constant objects.	tf
return_attention_mask	Whether to return the attention mask.	True
is_split_into_words	Whether or not the input is already pre-tokenized (e.g., split into words). If set to True, the tokenizer assumes the input is already split into words (for instance, by splitting it on whitespace) which it will tokenize. This is useful for NER or token classification.	True
verbose	Whether or not to print more information and warnings.	True

Four variations of the experiment were conducted for one of the emotions, anger. By comparing the performance of all the experiments, the best architecture for building the regressor models was finalized for the remaining emotions. The different variations of BERT for regression on anger tweets are shown in Table 6. Here, BERTv1 is the Bert-base-model which is trained for 3 epochs; BERTv2 corresponds to the Bert-base-model fine-tuned with two additional Dense *'elu'* layers of output shapes (None, 138) and (None, 28) respectively,

and a dropout layer of value 0.1. This model variation is trained for 3 epochs; BERTv3 has a similar architecture as BERTv2 but it makes use of '*relu*' activation in the Dense layers and is trained for 5 epochs; BERTv4 is similar to that of BERTv3. However, it is trained for 10 epochs. All the BERT variations for the regression task make use of only one output node with a sigmoid activation function in the output layer which generates the intensity value of the tweet.

Table 6: BERT Regressor's experimental variations for anger tweets.

BERT configurations	**Testing MSE**
BERTv1	0.0179
BERTv2	0.0180
BERTv3	0.0148
BERTv4	0.0131

Out of these variations, BERTv4 has performed the best with a low MSE value, and hence this architecture of BERT regressor was employed for building the remaining emotion regressors. Table 7 shows the regressors built for each of the emotions along with their MSE values using the BERTv4 configuration.

Table 7: Evaluation of BERTv4 regressor on anger, fear, sadness, and joy emotions.

	Anger	**Fear**	**Sadness**	**Joy**
Testing MSE	0.0131	0.0155	0.0154	0.0219

2.3.4.2. Classification

The BERT model for the classification task also accepts the same kind of inputs as that of a regressor such as *input_ids* and *attention_masks*. However, when generating the inputs for the model using the tokenizer function, the "*max_length*" argument of Table 4 is set to 44. For the classification task the features "*tweet id*" and "*intensity*" are irrelevant and have been dropped. Hence, when training the BERT model only the "*text*" sentence is considered as an input feature and "*emotion*" as the output feature. As the output label is categorical, the emotions have been converted to numerical using the *LabelEncoder* function. The values in the "*emotion*", anger, fear, joy, and sadness are now represented as 0, 1, 2, and 3 respectively.

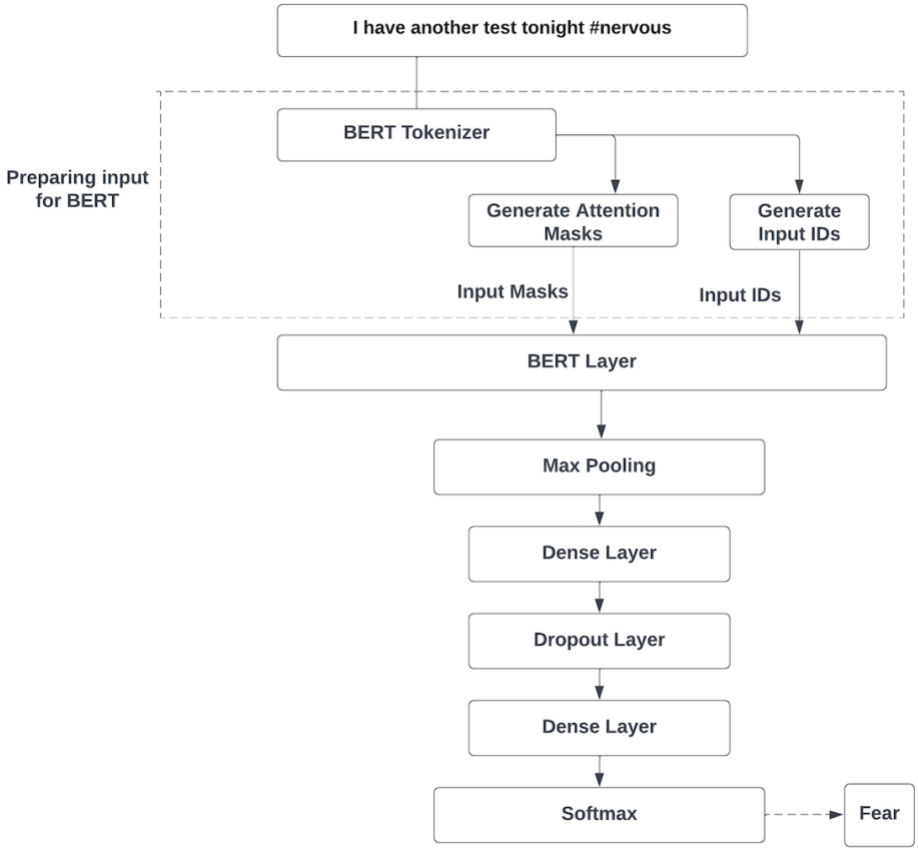

Figure 2: BERTv4 model summary for the emotion classification task.

The classification model was built with the BERTv4 configuration. The output layer consists of four nodes which represent the four output classes. The "*softmax*" activation function was used on the output nodes to get the probabilities of each tweet corresponding to each class. The architecture diagram for predicting the type of emotion using BERTv4 configuration with four epochs is shown in Figure 2. The example tweet "*I have another test tonight #nervous*" when sent as an input to model as shown in Figure 2 will yield an output "*Fear*".

Results and Evaluation

Mohammad et al [1] organized a shared-task in 2017 that focused on intensity prediction on tweets. The authors discuss the competition conducted for building regressors for the dataset that they have developed [10]. The participants were free to build a regressor from scratch and were allowed to combine training and validation sets for the purpose of training the regressors. The participants' models were evaluated against the gold rating by calculating the Pearson Correlation Coefficient. The script for evaluating the predictor model is available in [12]. The Pearson correlation coefficient for all four regressors of the 22 teams that participated in the WASSA-2017 competition against the result of the thesis is shown in Table 8.

Two experiments were conducted using BERTv4. In the first experiment, the input to the regressor is the actual sentence concatenated with the actual emotion. Here, a [SEP] token is introduced between the tweet and the generated sentence from the emotion to distinguish that these are separate entities. This newly created processed sentence is now converted to embeddings which are passed as inputs to the BERT regressors. The first experiment makes use of the BERTv4 configuration. The evaluation results of this experiment are shown in Table 9. In the second experiment, the tweet without the concatenation of the emotion was treated as input to the BERTv4 model to showcase the importance of combining the emotion with the actual text sentence during regression. The results show that there was a slight decrement in performance as shown in Table 9.

Table 8: Pearson correlation coefficients (r) and ranks (in brackets) obtained by the participants on the full test sets.

Team Name	r avg. (rank)	r fear (rank)	r joy (rank)	r sadness (rank)	r anger (rank)
1. Prayas	0.747 (1)	0.732 (1)	0.762 (1)	0.732 (1)	0.765 (2)
2. IMS	0.722 (2)	0.705 (2)	0.726 (2)	0.690 (4)	0.767 (1)
3. SeerNet	0.708 (3)	0.676 (4)	0.698 (6)	0.715 (2)	0.745 (3)
4. UWaterloo	0.685 (4)	0.643 (8)	0.699 (5)	0.693 (3)	0.703 (7)
5. IITP	0.682 (5)	0.649 (7)	0.713 (4)	0.657 (7)	0.709 (5)
6. YZU NLP	0.677 (6)	0.666 (5)	0.677 (8)	0.658 (6)	0.709 (5)
7. YNU-HPCC	0.671 (7)	0.661 (6)	0.697 (7)	0.599 (9)	0.729 (4)
8. TextMining	0.649 (8)	0.604 (10)	0.663 (9)	0.660 (5)	0.668 (10)
9. XRCE	0.638 (9)	0.629 (9)	0.657 (10)	0.594 (10)	0.672 (9)
10. LIPN	0.619 (10)	0.58 (11)	0.639 (11)	0.583 (11)	0.676 (8)
11. DMGroup	0.571 (11)	0.55 (12)	0.576 (12)	0.556 (12)	0.603 (11)
12. Code Wizards	0.527 (12)	0.465 (16)	0.534 (15)	0.532 (14)	0.578 (13)
13. Todai	0.522 (13)	0.470 (15)	0.561 (13)	0.537 (13)	0.520 (16)
14. SGNLP	0.494 (14)	0.486 (14)	0.512 (16)	0.429 (18)	0.550 (14)
15. NUIG	0.494 (14)	0.680 (3)	0.717 (3)	0.625 (8)	-0.047 (21)
16. PLN PUCRS	0.483 (16)	0.508 (13)	0.460 (19)	0.425 (19)	0.541 (15)
17. H.Niemtsov	0.468 (17)	0.412 (17)	0.511 (17)	0.437 (17)	0.513 (17)
18. Tecnolengua	0.442 (18)	0.373 (18)	0.488 (18)	0.439 (16)	0.469 (18)
19. GradAscent	0.426 (19)	0.356 (19)	0.543 (14)	0.226 (20)	0.579 (12)
20. SHEF/CNN	0.291 (20)	0.277 (20)	0.109 (20)	0.517 (15)	0.259 (19)
21. deepCybErNet	0.076 (21)	0.176 (21)	0.023 (21)	-0.019 (21)	0.124 (20)
Late submission					
* SiTAKA	0.631	0.626	0.619	0.593	0.685
Our Thesis Result					
BERTv4	**0.7654**	**0.7885**	**0.7334**	**0.7942**	**0.7483**

Table 9: Evaluation results of BERTv4, and BERTv4 without combining emotion with text.

Evaluation metrics	Models	Anger	Fear	Joy	Sadness	Avg
Pearson correlation coefficient on full test set	**BERTv4**	0.7483	0.7855	0.7334	0.7942	0.7654
	BERTv4 without combining the emotion with text.	0.7260	0.7664	0.7351	0.7798	0.7518
Pearson correlation coefficient on those sentences in the test set where intensity scores >= 0.5	**BERTv4**	0.5996	0.5935	0.4768	0.6228	0.5732
	BERTv4 without combining the emotion with text.	0.5982	0.5620	0.5262	0.5930	0.5699

The classifier model has yielded an accuracy of 84.15% on the test dataset. It can be seen from the confusion matrix of Figure 3 that most of the misclassifications revolve between the "*fear*" and "sadness" classes.

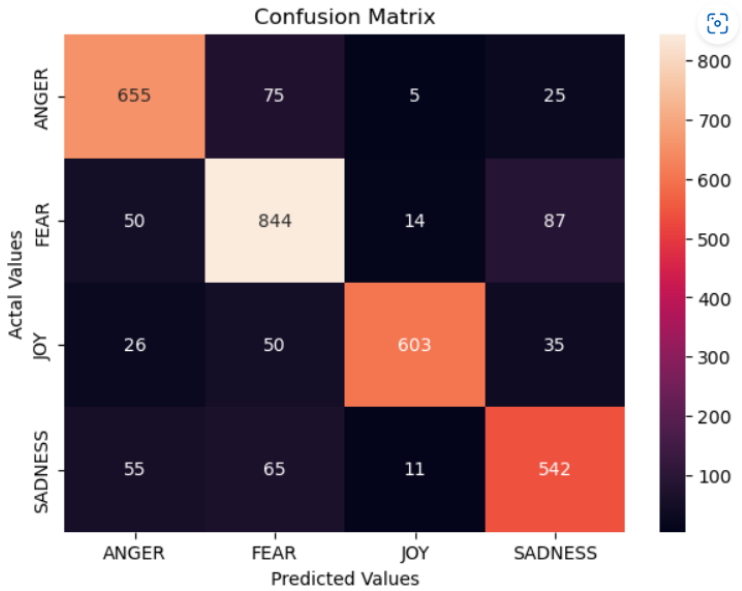

Figure 3: Confusion matrix of the BERTv4 classifier on the test set.

Another experiment was performed where the classifier and the regressor were working as a whole system.

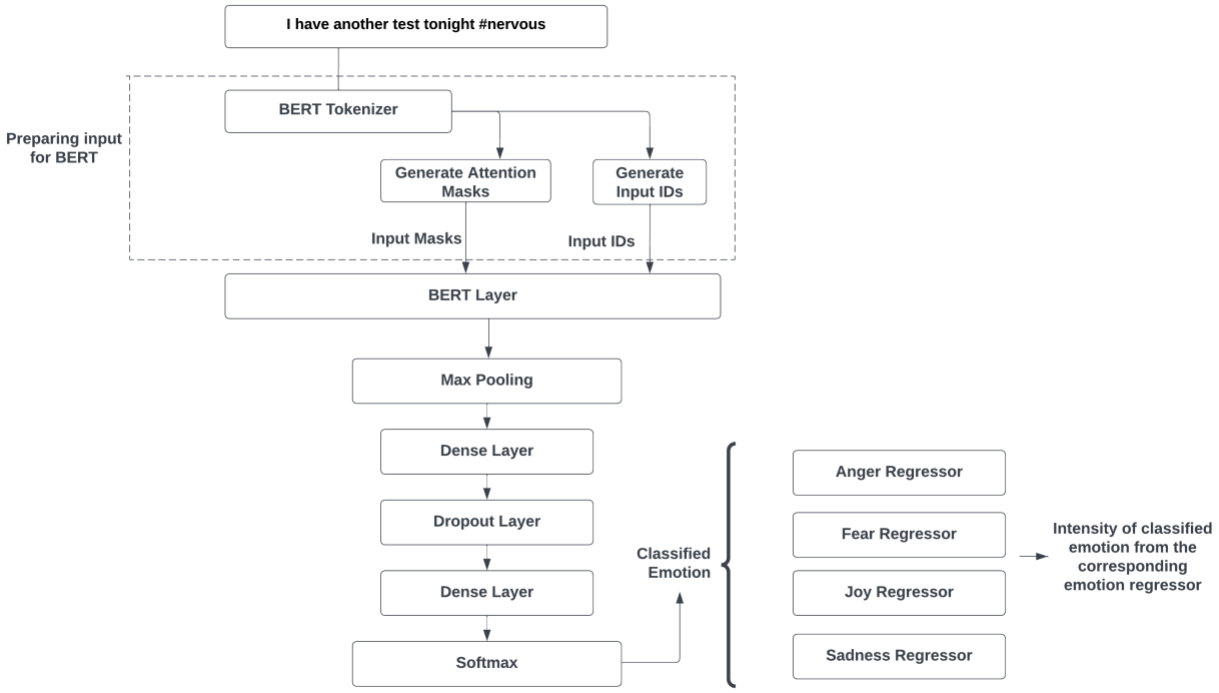

Figure 4: Classifier concatenated with regressor system.

This system accepts a tweet and performs classification to identify the type of emotion the tweet projects. Then based on the type of emotion, the tweet, and the classified emotion are sent to the designated regressor.

Table 10: Evaluation results of the classifier concatenated with regressor models.

	Anger	**Fear**	**Joy**	**Sadness**	**Avg**
Pearson correlation coefficient on full test set	0.3830	0.5860	0.6317	0.5023	0.5250

For example, if the tweet is classified as anger, then the tweet along with the emotion of *"anger"* is sent to the anger regressor to predict the intensity. If a tweet is misclassified, then its intensity is set to zero. The architecture diagram of this system is shown in Figure 4. The performance of the regressor in the architecture shown in Figure 4 is evaluated using the Pearson Correlation coefficient. It is observed from Table 10 that the average score is 0.53. This is because we are assigning zero as the intensity value if a particular emotion is misclassified.

Conclusion

In this work, several experiments were conducted to arrive at the best system for emotion classification and intensity prediction. The intensity-predicting regressor outperformed the state-of-the-art regressor which has been evaluated using the competition evaluation metrics. Such a system has several applications in detecting the emotion of individual users which can be utilized in developing mental health counseling bots, customer satisfaction in business needs, and improving the social status of the public. In the future, the research could be extended to improve the performance by collecting more data that supports other emotions than the primary ones.

REFERENCES

[1] Mohammad, S. M., & Bravo-Marquez, F. (2017). WASSA-2017 shared task on emotion intensity. arXiv preprint arXiv:1708.03700.

[2] Nazarenko, D., Afanasieva, I., Golian, N., & Golian, V. (2021). Investigation of the Deep Learning Approaches to Classify Emotions in Texts. In COLINS (pp. 206-224).

[3] Hasan, M., Rundensteiner, E., & Agu, E. (2014, May). EMOTEX: Detecting Emotions in Twitter Messages. 2014 ASE BIGDATA. In Socialcom/Cybersecurity Conference.

[4] Abdul-Mageed, M., & Ungar, L. (2017, July). Emonet: Fine-grained emotion detection with gated recurrent neural networks. In Proceedings of the 55th annual meeting of the association for computational linguistics (volume 1: Long papers) (pp. 718-728).

[5] Purver, M., & Battersby, S. (2012, April). Experimenting with distant supervision for emotion classification. In Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics (pp. 482-491).

[6] Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In Proceedings of ACLIJCNLP 2009

[7] Bollen, J., Mao, H., & Pepe, A. (2011). Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena. In Proceedings of the international AAAI conference on web and social media (Vol. 5, No. 1, pp. 450-453).

[8] R. Fan, K. Chang, C. Hsieh, X. Wang, and C. Lin, "Liblinear: A library for large linear classification," The Journal of Machine Learning Research, vol. 9, pp. 1871–1874, 2008.

[9] Alm, C. O., Roth, D., & Sproat, R. (2005, October). Emotions from text: machine learning for text-based emotion prediction. In Proceedings of human language technology conference and conference on empirical methods in natural language processing (pp. 579-586).

[10] Mohammad, S. M., & Bravo-Marquez, F. (2017). Emotion intensities in tweets. arXiv preprint arXiv:1708.03696.

[11] https://developers.google.com/machine-learning/crash-course/classification/precision-and-recall

[12] http://saifmohammad.com/WebPages/EmotionIntensity-SharedTask.html

[13] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep

bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

[14] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30.

[15] Strapparava, C., & Mihalcea, R. (2008, March). Learning to identify emotions in text. In Proceedings of the 2008 ACM symposium on Applied computing (pp. 1556-1560).

[16] Thelwall, M., Buckley, K., Paltoglou, G., Cai, D., & Kappas, A. (2010). Sentiment strength detection in short informal text. Journal of the American society for information science and technology, 61(12), 2544-2558.

[17] Jurgens, D. (2013, June). Embracing ambiguity: A comparison of annotation methodologies for crowdsourcing word sense labels. In Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (pp. 556-562).

[18] Kiritchenko, S., Zhu, X., & Mohammad, S. M. (2014). Sentiment analysis of short informal texts. Journal of Artificial Intelligence Research, 50, 723-762.

[19] https://medium.com/analytics-vidhya/what-is-a-confusion-matrix-d1c0f8feda5