Improving mental disorder detection through emotion recognition

Mihail Bogdan Chirobocea, Răzvan George Costea

University of Bucharest

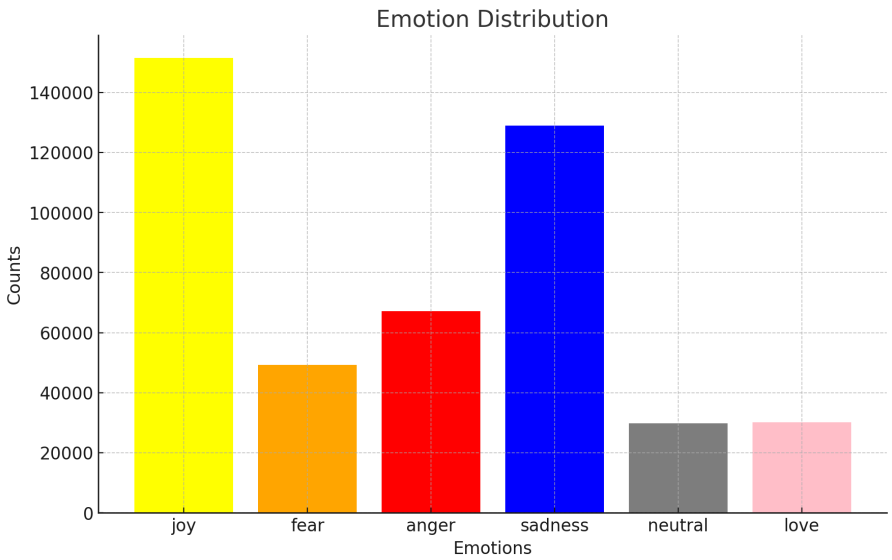
1. Dataset

In our approach we targeted two kinds of datasets.The first one consists of texts along with a label of a single emotion associated with it, and a second one that consists of texts along with a label that tell us if the person has depression or not.

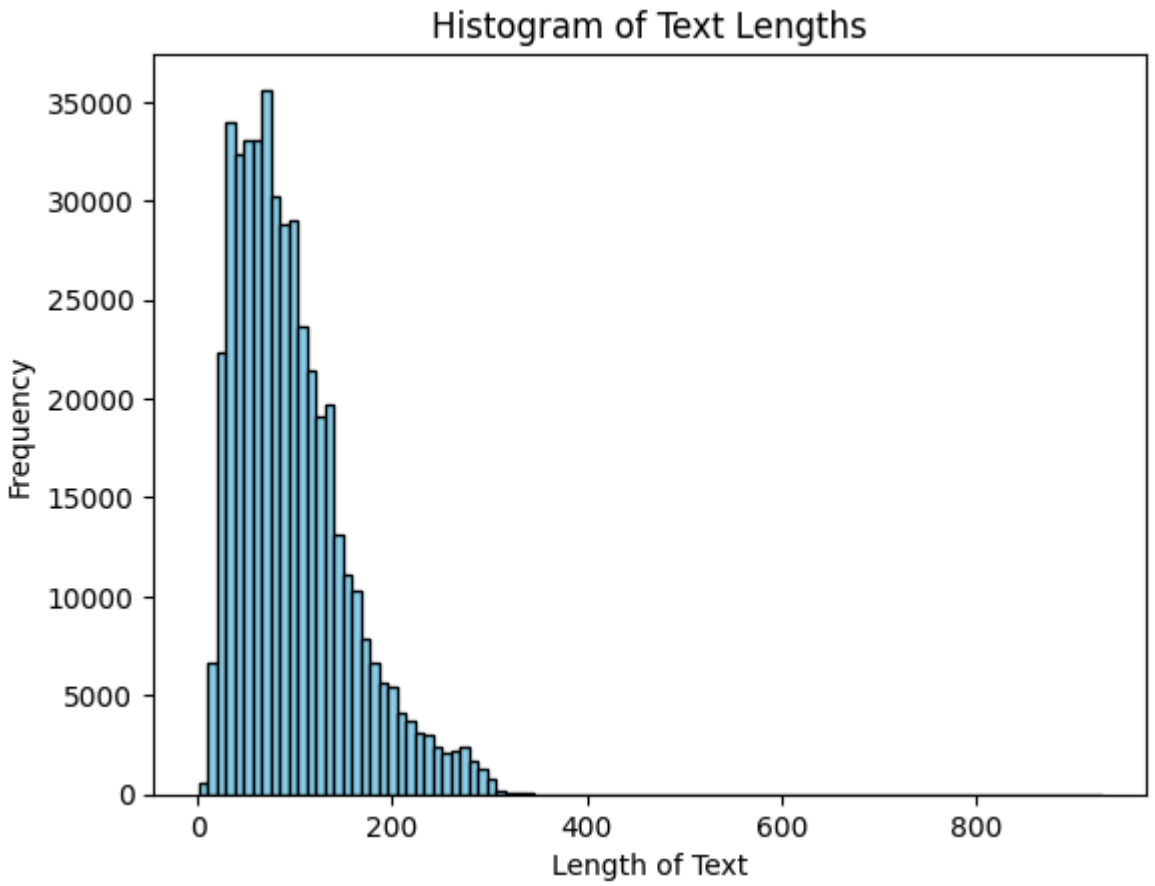
a. The emotion dataset is a combination of seven publicly available datasets, namely:
- Emotion Classification NLP
- Emotion Dataset
- Emotion Detection from Text
- Emotions
- Emotions in text
- Go Emotions: Google Emotions Dataset
- Sentiment & Emotions Labelled Tweets

All of them are human written texts from the internet and manually labeled by people, with no use of synthetic data.

After this merging, we ended up with a total of 33 distinct emotions. To make sure we have enough data for each class and remove a very strong imbalance between classes, we setted a threshold of minimum data per class and keep only the following classes: joy, fear, anger, sadness, neutral, love.



Nevertheless we made sure that there are no duplicates in our text data. FInally we visualized the length text distribution and noticed that it's similar to a Gaussian distribution,as we expected, with very few exceptions that we excluded from our final dataset.

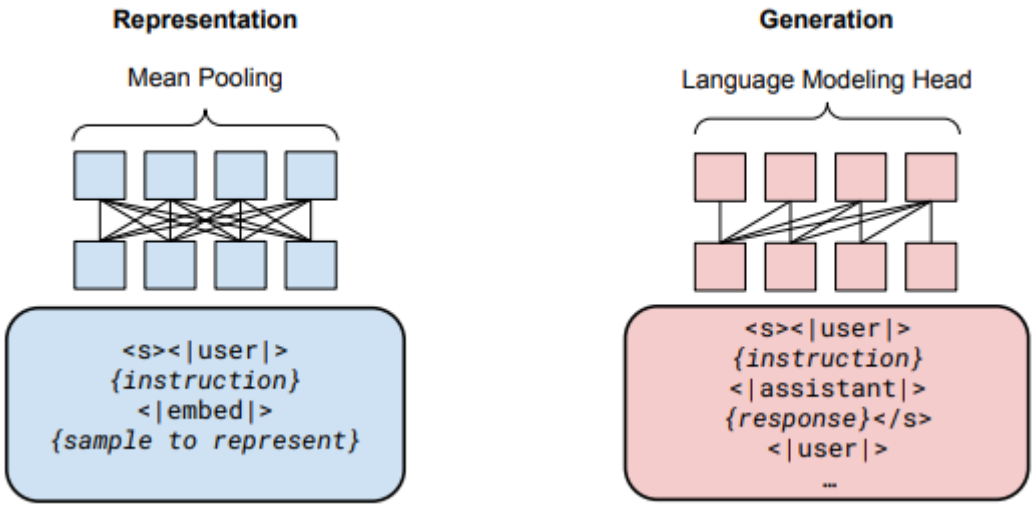


After all this filtering, we ended up with a dataset of about 450k samples.

b. For the second dataset, we used Sentimental Analysis for Tweets which consist of about 10k samples from tweets, where 8k of them are labeled with no depression and 2k are labeled with depression.

2. Model

As a feature extractor we choose to use a 7B LLM adapted to classification task. Namely, we used GritLM 7B which is based on Mistral 7B. The model used exactly the same feed forward modules of the original model, while for the attention they duplicate each attention block, and change one of them from the causal attention of the original model to the bi-directional attention which is well suited for feature extraction.



The bi-directional heads are further fine tuned on specific encoding tasks using contrastive learning methods.

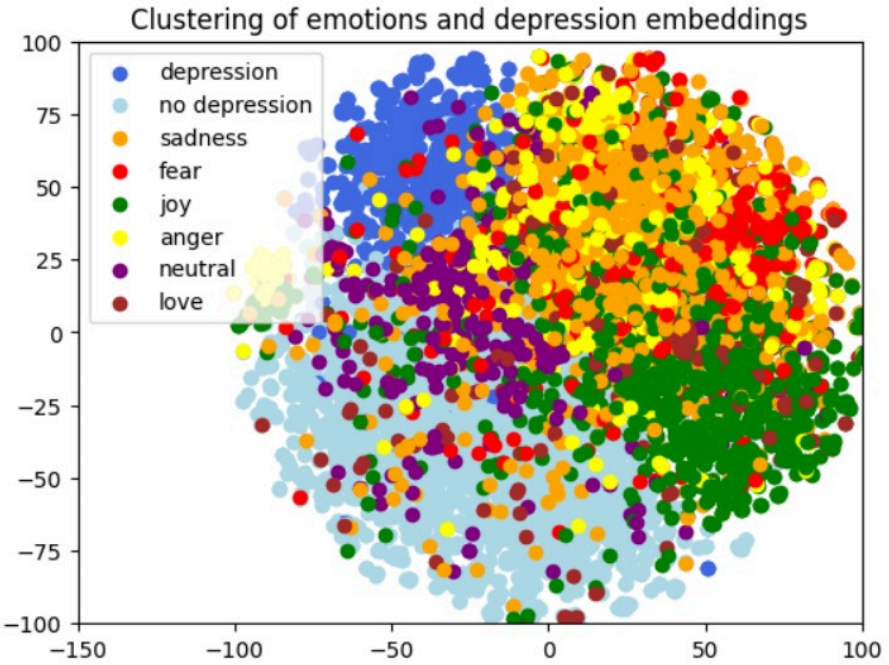
This model takes advantage of a huge amount of parameters of an LLM compared to other encoder models and also makes use of the diversity of the pretrained data, as so, being more robust to typoes for example, while it make sure to use the bi-directional attention which is crucial for the quality of embeddings targeting classification tasks.

Finally, we encoded each token and averaged all of them on the encoding dimension, ending up with a vector of 4096 per sample.

3. Feature visualizations

To have a better understanding on why using knowledge about emotion is worth trying when our main task is to detect several mental health disorders we need to visualize the features that were extracted using the chosen transformer encoder. Thus, we can see if the emotion and depression features are in the same neighborhood and, more fascinating, we can see what kind of emotion is closer to the depressive condition.

For this we need to reduce the dimension of our data from 4096 to 2 and plot the points on a 2D plane, coloring each one of them with a specific color based on the emotion or depression classes. We chose t-SNE method for dimensionality reduction and randomly sampled 10000 embedding vectors from each dataset (emotion and depression). After t-SNE we get 2D vectors that can be plotted and visualized.



It can be observed that sadness, anger and fear data points are more close to depression than to non-depression. Also, joy is on the other side of depression and is close to non-depression. Neutral is exactly in the middle of the plot, which makes perfect sense. Therefore, it is worth trying to predict depression out of a pre-trained emotion recognition model.

4. Emotion detection

Using the previous encoded features, we constructed a MLP like head of classification. This network consists of three MoE (Mixture of Experts) layers and a final output layer.

Each MoE layer is constructed like the following. Firstly we have to set the number of experts, which is a hyperparameter, let's call it n. After that we have a gating layer, which is a fully connected layer from the input size to the size n. This layer is followed by a softmax function, as so, the aim of this layer is to decide the weight of each expert based on the input. In the same time, the input is passed to n experts, each expert is a typical fully connected layer. The outputs of those experts are weighted by the gating values and summed up to a final MoE layer output and passed further to the network. Note that all this process can be optimized via gradient descent. The idea of these so-called MoE layers is inspired by Mixtral.

For optimizations and strong generalizations we used the following:

- Weighted cross entropy loss to overcome the unbalanced classes
- Label smoothing to improve generalization
- Dropout on input features to act like masking, in order to improve generalization and remove biases
- Dropout between MoE layers to improve generalizations
- Weight decay to overcome overfitting
- Noise and contrast augmentations to improve generalizations
- Cosine Scheduler to overcome both overfitting and local minimum issues

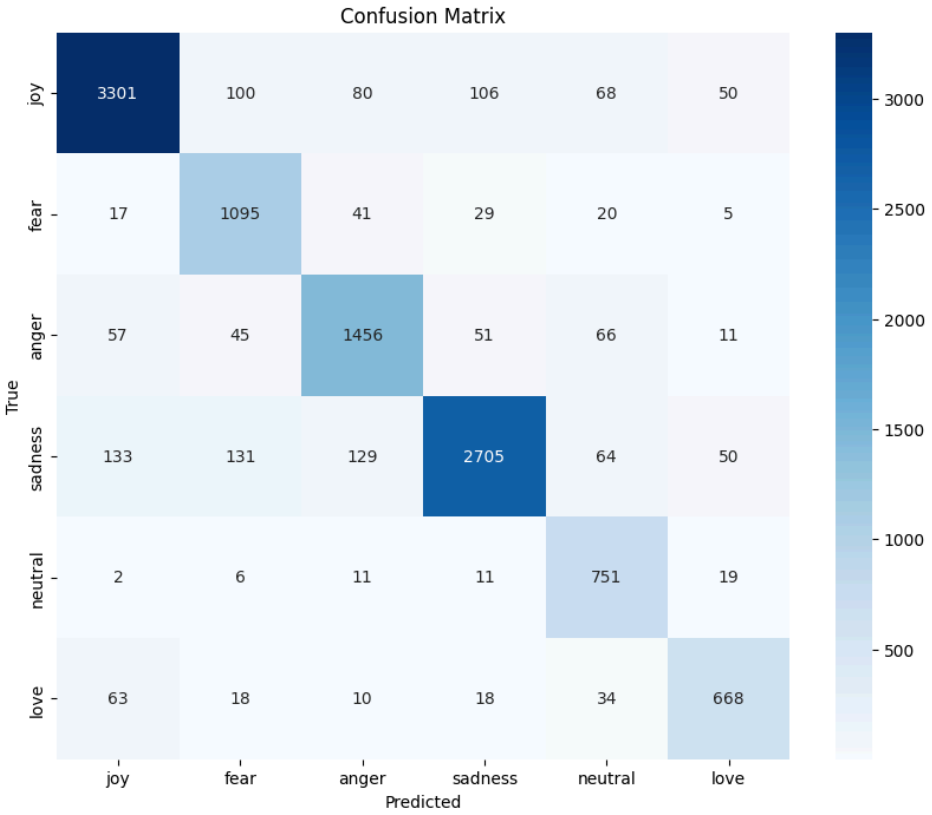
We used a train (433k samples), validation (11k samples) and test (11k samples) split, where we used a validation subset to choose the best model and test subset to report our final metrics.

Our best metrics were gained with the following hyperparameters:

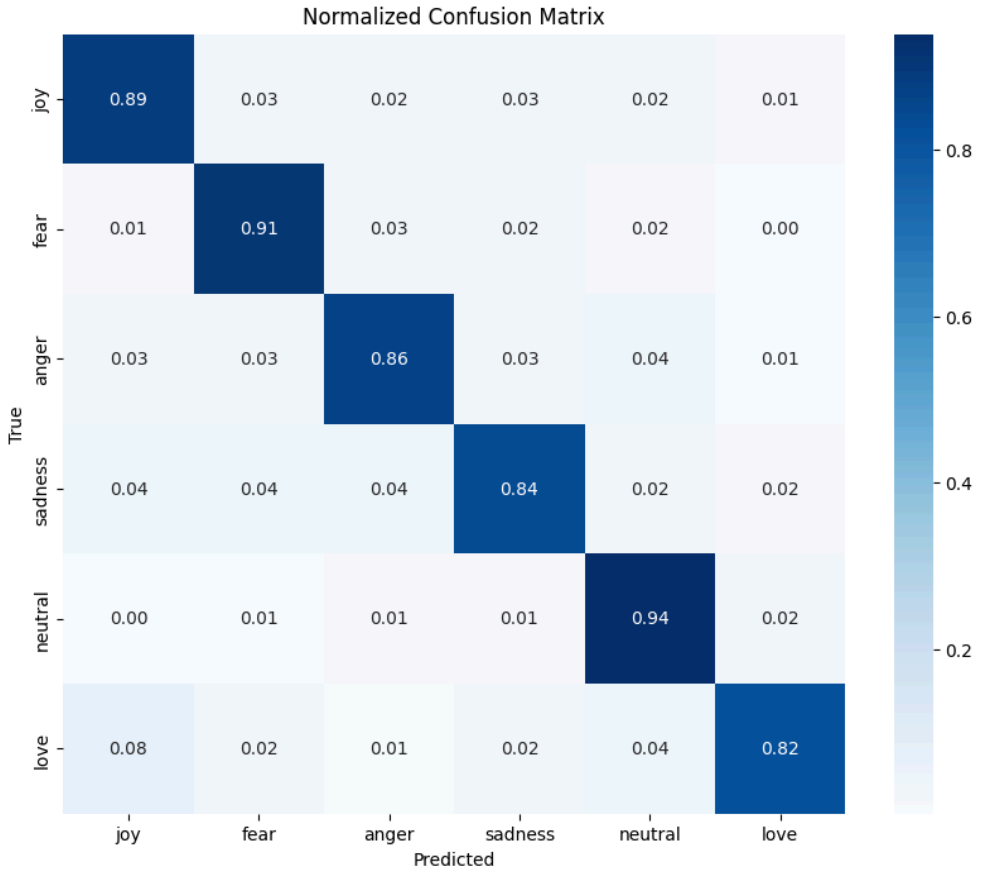
- AdamW optimizer with a peak learning rate of 1e-4 for 12 epochs
- Batch size of 1024
- Hidden size of MoE layer of 8192, 256 and 32 respectively
- GELU activations and Layer Norm

The metrics on test subset are the following:

- Macro f1: `0.86`
- Weighted f1: 0.87



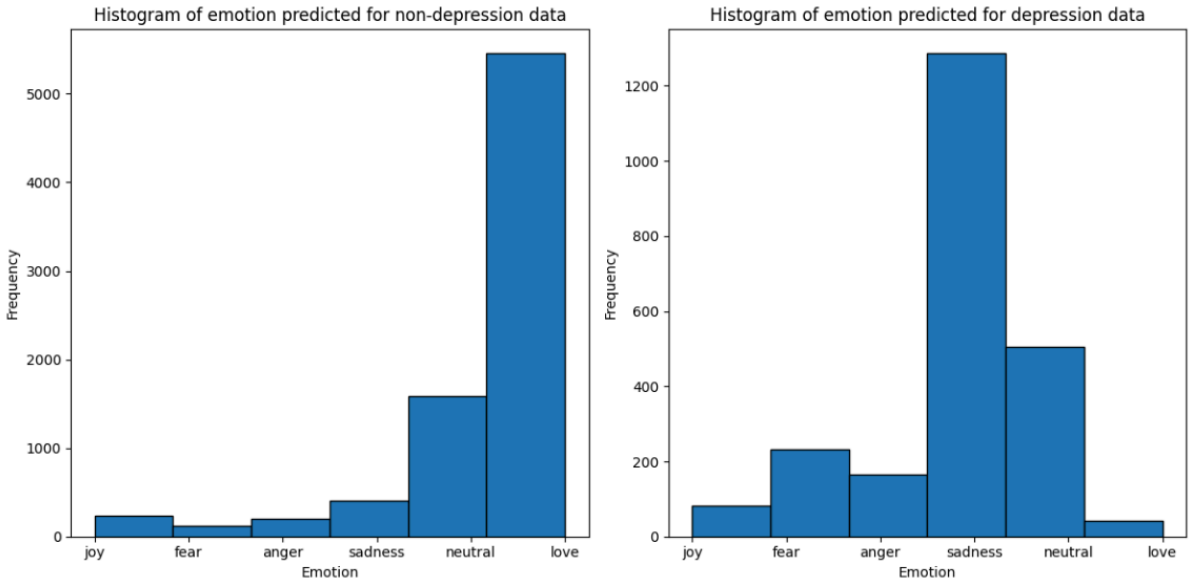
For a better view we can take a look at the normalized confusion matrix.



We can see that the most mis-predicted case is that of love emotions confused with joy, which is indeed a fairly similar emotion.

5. Depression detection

First of all, we want to see what emotions are predicted by the model for depression data. For that, simply just feed the depression data to the model and separate the predicted emotions for both depressive condition and non-depressive condition. The output is shown in the following histogram.



It can be observed that the model with emotion knowledge can predict depression related emotions seen in the visualization section. In the depressive condition the top 3 emotions that are felt, from the neural network point of view, are sadness, neutral emotion, fear. When a person is not depressed, there is love predicted as the most dominant emotion.

Now we want to predict mental disorders. The trained MLP from the previous section, except the output layer, is used for fine tuning on depression data. The model is a network with two output classes, one for depression class and one for non-depression class. In this step we use the knowledge from emotion classification to detect depressive conditions.

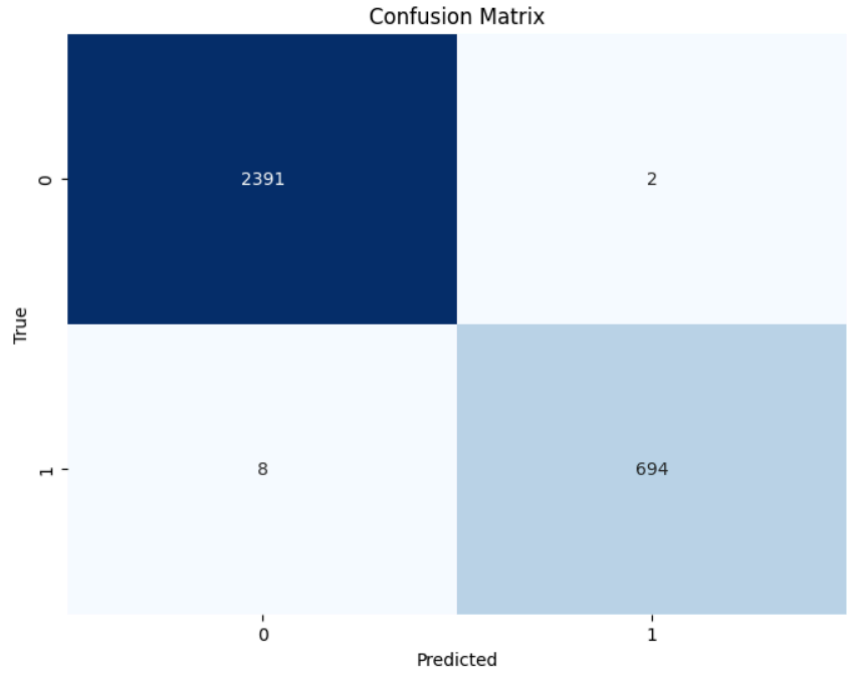
For optimizations and strong generalizations we used the same approach described in the emotion detection section.

We used a train (4125 samples), validation (3094 samples) and test (3095 samples) split. Our best metrics were gained with the same hyperparameters as in the emotion case, except the epochs where for the depression model we only trained for 5 epochs and got a very high accuracy.

The metrics on test set are the following:

- Macro f1: 1.0
- Weighted f1: 1.0

The confusion matrix for the test set:



6. Future work

Nevertheless, all the data might suffer from partial duplicate texts that might bias the model towards some kind of words. To prevent that, Rouge filtering might be employed, which is an O(n*n) complexity algorithm.

To prevent miss-labeled data, we can further take use of training dynamics to identify potential wrong data and either correct it or remove it.

Furthermore, testing this approach on different disorder diseases is another thing that can sustain the powerfulness of our approach to boost detection for edge disorders that typically does not have a sufficient amount of data.