Social Media Mental Health Disorder Classification

Alex-Mihai Serafim* Group 407
Andrei Călin* Group 411
David-Cosmin Bejenariu* Group 407
Ionuț-Valentin Bălu* Group 411

***Abstract*—The analysis of emotions expressed in social media plays a crucial role in understanding psychological disorders and related phenomena. This study utilizes a dataset sourced from Reddit, focusing on posts related to mental health. The dataset consists of 5,607 posts categorized into five classes representing various mental health disorders, providing valuable insights into how individuals articulate their emotional experiences online.**

Findings reveal significant nuances in language use across different disorder categories, underscoring the potential of social media data in enhancing diagnostic and predictive capabilities in mental health research. This approach not only contributes to advancing computational methods in psychology but also underscores the importance of digital platforms as sources of valuable behavioral data for understanding and addressing psychological phenomena.

I. Introduction and Related Work

The analysis of social media content to detect mental health issues has become a significant area of research. Social media platforms offer a wealth of user-generated content that can provide insights into users' mental states.

A study published in BMC Psychology [1] examined Reddit posts to identify various mental health conditions, demonstrating that computational techniques can effectively detect mental health issues from social media content . Similarly, the study "Quantifying Mental Health Signals in Twitter" [2] utilized NLP and machine learning to analyze tweets, finding that Twitter data could also reveal patterns indicative of mental health conditions .

In contrast, a study published by the National Center for Biotechnology Information [3] highlighted the limitations of relying solely on social media data for diagnosing mental health disorders. The researchers emphasized the necessity of integrating additional factors, such as clinical assessments and offline behaviors, for a more comprehensive understanding of an individual's mental health .

These studies collectively underscore both the potential and challenges of using social media data to detect mental health issues, providing a foundation for refining computational methods and integrating diverse data sources in future research.

II. Implementation Details and Results

A. Dataset and data processing

In our experiments, we will be using the Kaggle Reddit Mental Health dataset containing over 5000 social media posts from Reddit. Our analysis will solely rely on the texts from the posts, written in English, where each text can represent a sign of one of the following psychological disorders: Stress (0), Depression (1), Bipolar disorder (2), Personality disorder (3) and Anxiety (4). The final distribution of the dataset is shown in Fig. 1. We can see that the dataset is well distributed.

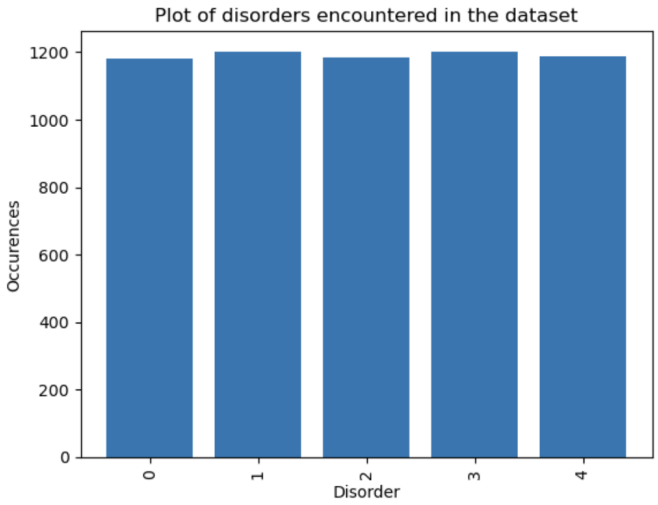

Fig. 1. Dataset distribution (0-Stress, 1-Depression, 2-Bipolar disorder, 3-Personality disorder, 4-Anxiety)

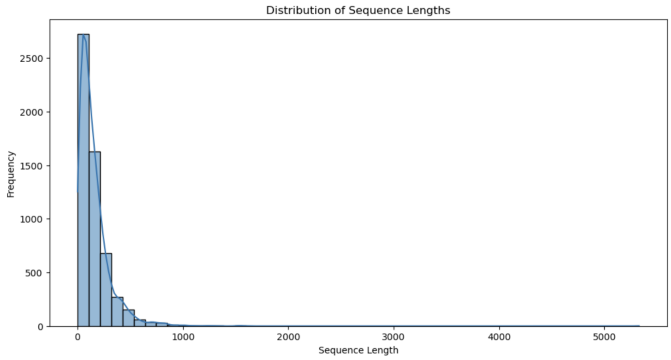

Fig. 2. Sentence length distribution

B. Experiments

a) Classical machine learning algorithms:

Pre-processing

Data preprocessing is a critical step that involves transforming raw data into a clean and usable format suitable for modeling. When working with text data, this process is especially important due to the unstructured nature of text. The

processing involves various techniques to clean, transform, and prepare data for analysis.

The first step taken, given the dataset at hand, was to concatenate the title and the text for each entry, since both elements could contain essential clues and words in determining the right label.

After joining the text and the title for each entry, we proceeded with cleaning the data, in our case meaning that we removed all of the null entries that were useless for our model, since they contained no information that could help with the classification task. There were just 29 null entries, so we still had plenty of data to work with.

In order to extract features from the text data we created an extractor class that had the role of transforming each phrase into proper values that could be given to a model.

The class had the following capabilities:

- replacement of emojis with actual text, using the demojize method from the emoji library. The scope of this method was to give a textual description of each emoji, because they could help us determine some specific sentiments when used in different contexts.
- removal of all of the digits, punctuation and characters that differed from white spaces and letters. The method was used for ”cleaning” the text, reducing it only to sequences of letters separated by a space.
- removal of the stop words present in the text, since they usually do not add a lot of value to the deeper meaning of a message;
- replacement of each word of the text with its lemma, since each word usually keeps its meaning, no matter the form in which it is transformed in the sentence.

The class also presented two methods that had the role of applying *CountVectorizer* and *TfidfVectorizer* (imported from *sklearn*) to the data, basically transforming a textual input into a vectorized one, that a model can digest and process.

Used algorithms

Since we found an extractor that gave us good results even with Naive Bayes, we tried to apply it to the text and then run some models that can usually be a little more precise when it comes to classification tasks.

Using the Random Forest classifier resulted in an accuracy of 0.84, meaning that it was able to determine the right label for each entry with a lower error compared to the Naive Bayes. Figure 2 shows the confusion matrix for this method:

We also used the SVM (Support Vector Machine) classifier on the same dataset and using the same extractor, but it performed worse compared to the other two models presented above, with an accuracy of 0.69 and the following confusion matrix shown in Figure 3. We can see that the SVM model had a tendency to predict inputs with the label 1 instead of finding out the actual true label of the inputs.

The next idea was to use some models from the gradient boosting family. The ones in question are Gradient Boosting Classifier, XGBoost and LightGBM.

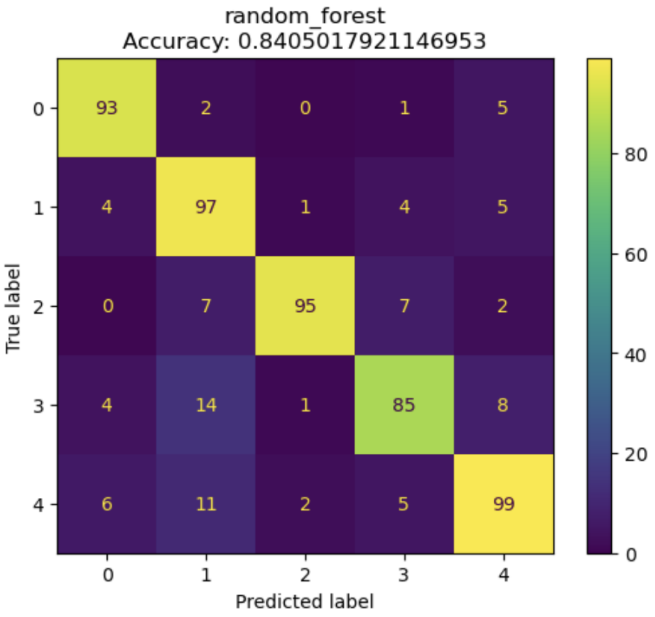

Fig. 3. Confusion matrix for our Random Forest approach

For the GBC which is an ensemble learning method that builds models sequentially, each one correcting the errors of its predecessor, we obtained an accuracy of 0.79. XGBoost, being an advanced implementation of the gradient boosting framework, with optimizations and regularization techniques, was considerably faster and it provided us with an accuracy of 0.82. But the one model that performed the best for our problem was Light Gradient Boost Machine. It uses histogram-based algorithms for faster computation and lower memory usage. We were able to obtain an accuracy score of 0.84. Results can observed in the following figures (Figure 4 - Figure 6).

b) First deep learning-based approach: The next experiment employs a deep learning approach, specifically a *recurrent neural network (RNN)*, to predict mental health disorders based on textual content from the Reddit posts. The model architecture includes an embedding layer for tokenization, *LSTM* layers for sequence processing, and *softmax* activation for multi-class classification. The study evaluates model performance using metrics such as accuracy and loss on a validation set.

To prepare the data for deep learning models, initial preprocessing steps were undertaken. Posts were tokenized using *Tensorflow Tokenizer* to break them down into individual words, ensuring consistency in format by padding sequences to a standardized length.

The core of the experiment involved constructing a deep learning model centered around a recurrent neural network (RNN), specifically incorporating *Long Short-Term Memory* (LSTM) cells. LSTMs are well-suited for analyzing sequential data like text due to their ability to capture long-range dependencies.

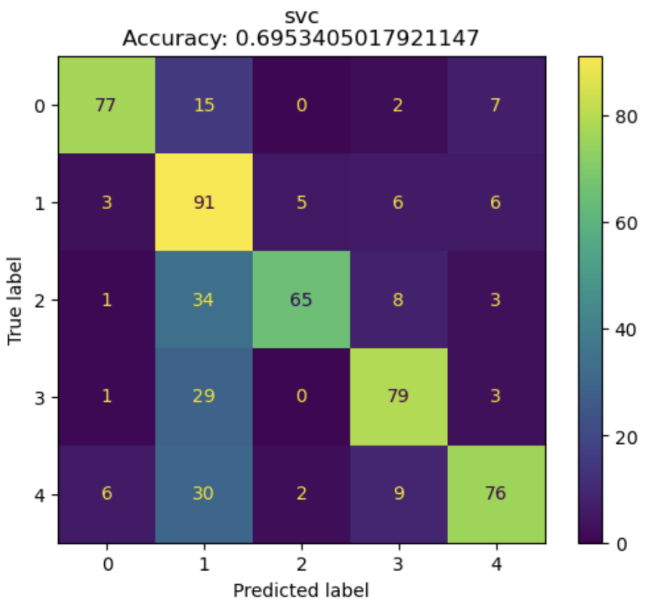

Fig. 4. Confusion matrix for our SVM approach

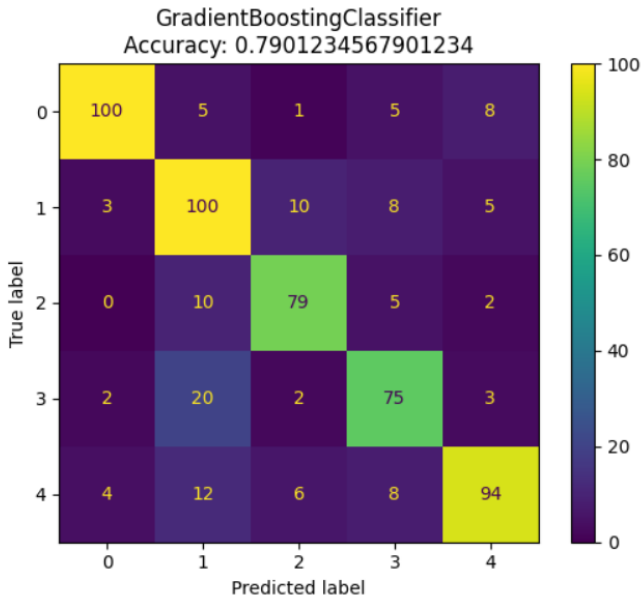

Fig. 5. Confusion matrix for our GradientBoostingClassifier approach

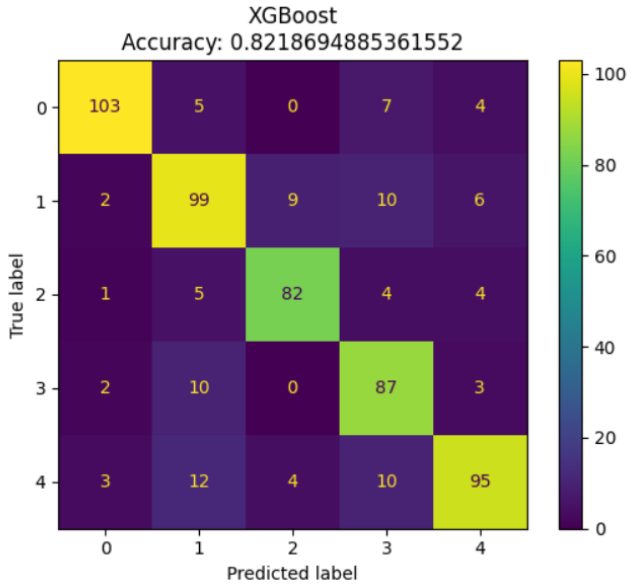

Fig. 6. Confusion matrix for our XGBoost approach

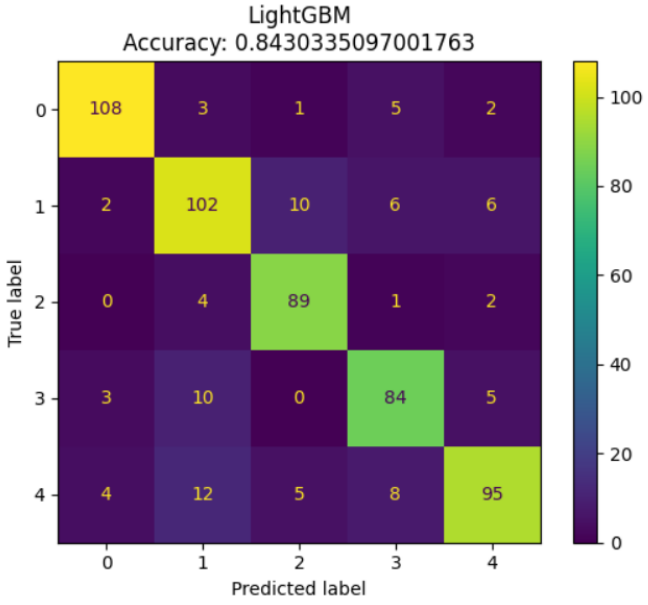

Fig. 7. Confusion matrix for our LightGBM approach

Within our RNN architecture, an embedding layer was applied as the initial processing step to convert words into numerical representations. This embedding facilitated the model's understanding of semantic relationships between words, critical for accurate classification. Instead of providing a pretrained embedding matrix, we opted for enabling the embedding layer to train.

The RNN was configured with multiple LSTM layers, each comprising 128 units and supplemented with dropout regularization to prevent overfitting. Dropout layers were strategically incorporated to enhance the model's generalization capability by randomly deactivating units during training.

Training of the RNN model utilized the Adam optimizer, which is efficient for handling large-scale datasets and dynamic learning rates. The model was evaluated using sparse categorical cross-entropy as the loss function, suitable for multi-class classification tasks where each post was assigned to one of the five predefined mental health disorder classes.

The training process spanned twenty epochs, with continuous monitoring of performance metrics on a validation set to ensure model robustness and prevent overfitting. Early stopping criteria based on validation performance were employed

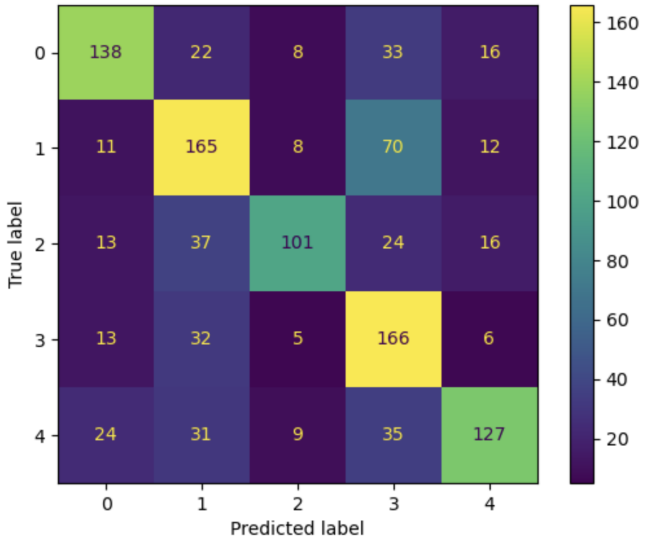

Fig. 8. Confusion matrix for our neural network approach

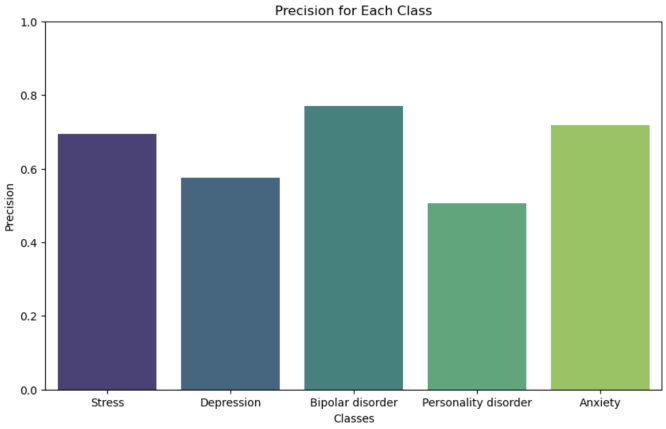

Fig. 9. Precision for our neural network approach

to optimize model performance.

The adjusted network did not exceed the expectation, performing with an accuracy of 63.55% and a loss of 1.7742.

c) Pre-trained model fine-tuned on lyric classification task:

BERT

We compared classical methods with a highly performant pre-trained BERT model, which is revolutionary in the sense that, unlike previous models, it bidirectionally trained the Transformer to better capture language nuances. The authors of the original BERT paper introduced the Masked Language Model training paradigm, where a subset of words is masked at input. Although BERT's initial objective was to reconstruct sentences, it proved highly successful on various downstream tasks.

Unlike classical methods, BERT does not require any pre-

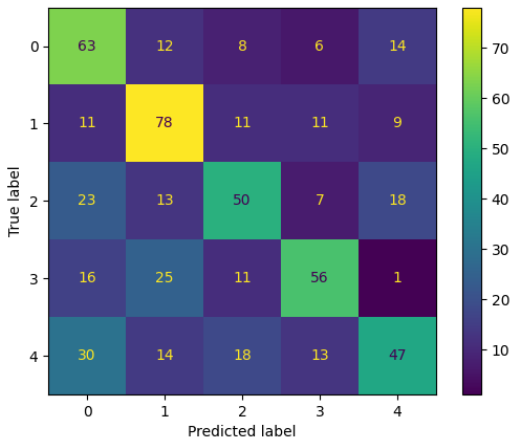

Fig. 10. Confusion matrix for pre-trained BERT.

processing of input, as it has its own method for handling data. Text is first tokenized, truncated, padded, and then transformed into embeddings as it passes through the network's layers. We used the lightweight and distilled DistilBERT model and tokenizer from the HuggingFace Transformers library. To prevent overfitting, we froze the intermediate layers during training, focusing only on the final classification layers. Training was done for 6 epochs due to computational constraints. This resulted in a test accuracy of 58.6%. Figure 10 shows that the model had an acceptable performance on all classes.

Integrated Gradients [4] We chose to focus on model interpretability for the BERT approach, since we obtained higher scores with other models. This provided some valuable insight into how BERT worked to classify the data and can help in understanding the particularities of the disorders.

To confirm and visualize the system's correctness, we employed feature attribution methods, which quantify the importance of each feature in making a prediction. One popular method for neural network architectures is Integrated Gradients. This method is based on two axioms: implementation invariance (two networks are equivalent if their outputs are equivalent) and sensitivity (if an input and a baseline differ in one feature and have different predictions, that feature must have a non-zero attribution).

The core idea behind Integrated Gradients involves taking a baseline input, a selected input from the dataset, and the target associated with that input. We iteratively transition from the baseline input to the selected input, constructing it one step at a time. At each iteration, we compute the integral of the gradients from our target to the features. This integration results in an attribution value for each feature, indicating the feature's contribution to the prediction.

In our case, the baseline input was an array of 0's and the [END] tokens, while the selected input consisted of the tokenized chosen text. We used the LayerIntegratedGradients class from the Captum library and extracted several represen-

tations for interpretation.

In 11, we can observe the activation of each word for a text piece corresponding to the depression class. Aside from the obvious activations such as ”depression”, we can see strong activations for phrases containing ”struggle”, ”ending it”, or ”anymore”, highlighting the specific tendencies of people suffering from depression.

For the anxiety example in 12, less obvious activations include ”school” (pointing to a source of anxiety), ”sleep”, ”breathe”, ”heart”, ”panic” (pointing to common symptoms associated with anxiety disorder such as panic attacks or insomnia).

According to the Integrated Gradients representation, we were able to extract meaningful representations of the disorders, which indeed helped with the classification task.

III. Conclusions and Further Improvements

We have experimented with various machine learning methods for classifying social media texts into possible psychological disorders. Our findings underscore the effectiveness of leveraging advanced natural language processing models like BERT and RNN, which captures intricate semantic nuances, alongside traditional machine learning algorithms like Naive Bayes and Random Forest. Moreover, the utilization of neural networks demonstrates their robustness in handling complex feature representations and achieving competitive performance in sentiment analysis tasks.

As a further improvement, we come to the realization that the deep learning approach would provide better results for larger datasets, thus by augmenting the dataset we expect to help the trained network perform better on this task.

By integrating these methodologies we pave the way for further exploration and refinement in the intersection of psychology and natural language processing.

References

[1] H. N. V. L. e. a. Tudehope, L., “What methods are used to examine representation of mental ill-health on social media? a systematic review.” *BMC Psychology*, 2024. [Online]. Available: https://rdcu.be/dLTkQ

[2] D. M. H. C. Coppersmith, G., “Quantifying mental health signals in twitter,” 2016. [Online]. Available: https://www.researchgate.net/publication/291365993_Quantifying_Mental_Health_Signals_in_Twitter

[3] G. Muscan, “Mental health analysis in social media posts: A survey,” *National Center for Biotechnology Information (NCBI)*, 2022. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9810253/

[4] M. Sundararajan, A. Taly, and Q. Yan, “Axiomatic attribution for deep networks,” 2017.

* Equal contribution.

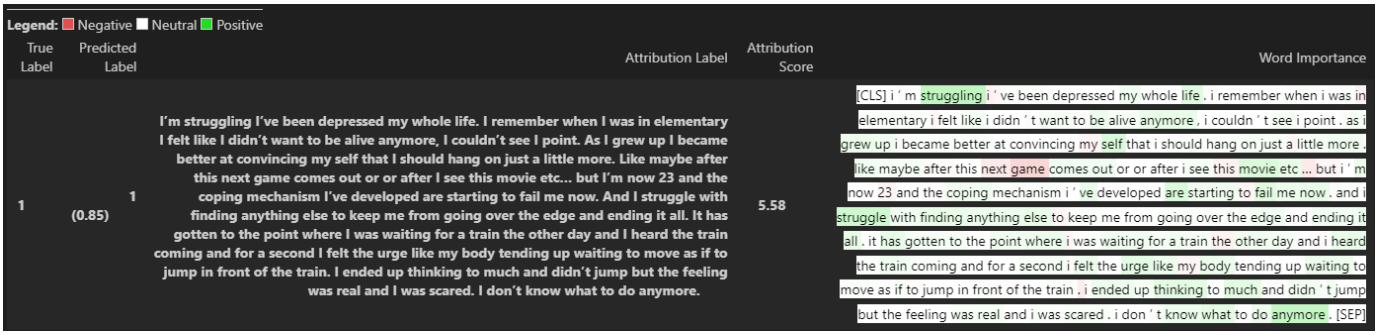

Fig. 11. Feature attribution with Integrated Gradients, depression example).

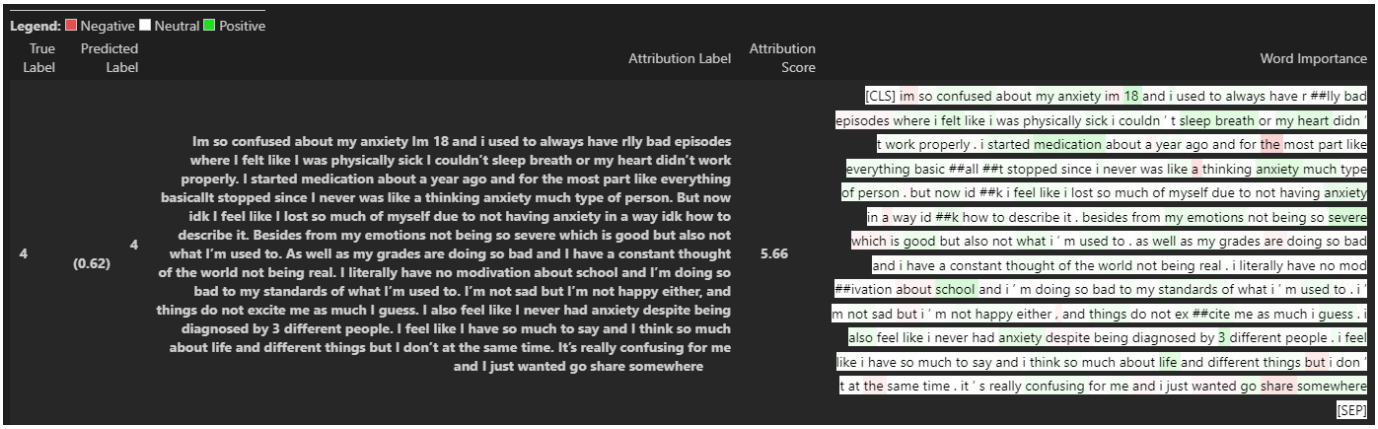

Fig. 12. Feature attribution with Integrated Gradients, anxiety example.

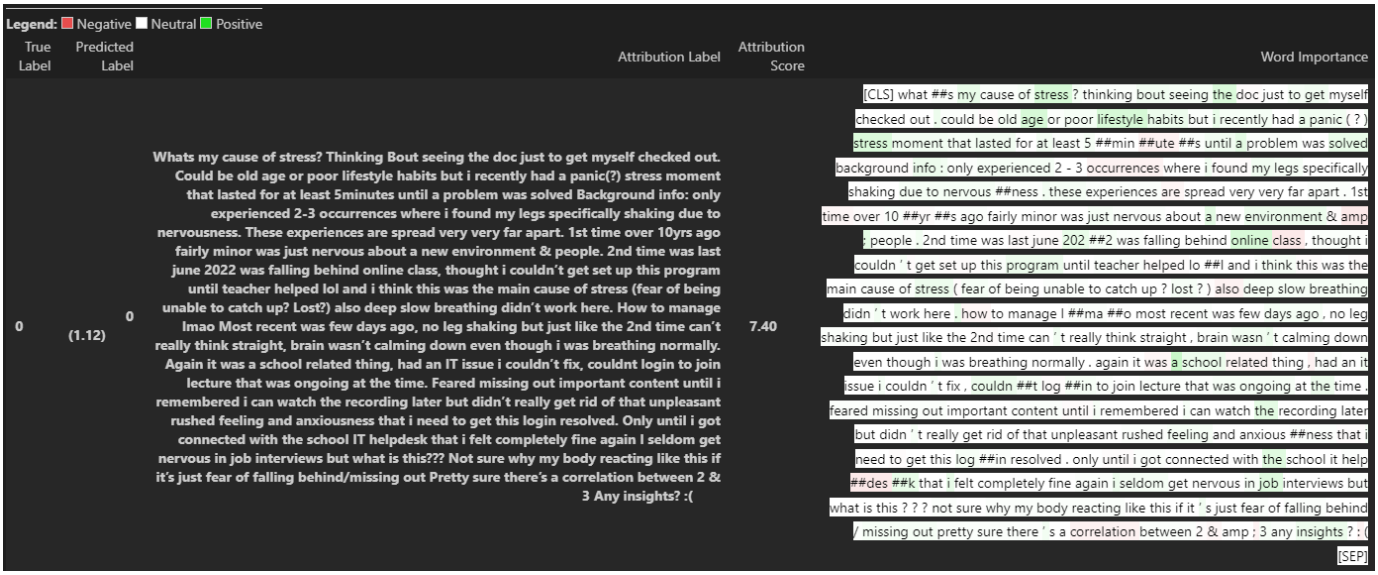

Fig. 13. Feature attribution with Integrated Gradients, stress example.