Comparative Analysis of Human Stress Prediction Using Multi-Models Natural Language Processing System

Mohammad Nasif Sadique Khan, A S M Nasim Khan, Md Fardin Rahman Ami, MD. Adnan Howlader,
Farah Binta Haque, Ehsanur Rahman Rhythm, Md Sabbir Hossain and Annajiat Alim Rasel
Department of Computer Science and Engineering (CSE)
Brac University
{a.s.m.nasim.khan, mohammad.nasif.sadique.khan, md.fardin.rahman.ami, md.adnan.howlader,
farah.binta.haque, ehsanur.rahman.rhythm, md.sabbir.hossain1}@g.bracu.ac.bd,
annajiat@gmail.com

***Abstract*—In the interconnected digital landscape of today, people have found various ways to express and share their stress online. Social media platforms, with their instantaneous reach, have become digital diaries where people unburden their stress. Status updates, tweets, and posts often bear the weight of their emotional struggles. Some lay their feelings bare with candid confessions, describing the anxieties and challenges they face. Others may resort to cryptic messages or metaphoric posts, using symbolism as a way to indirectly articulate their turmoil. For our research, we gathered a dataset of text from Reddit, a popular online platform known for its diverse discussions and conversations. Our aim was to analyze this text data to detect patterns and gain insights into the sentiments and emotions expressed by users. To achieve this, we employed both general machine learning algorithms and more advanced neural network algorithms. Using traditional machine learning techniques, we applied algorithms, such as Naive bayes, support vector machines, decision trees, random forests, etc. In parallel, We trained neural network models like BERT, LSTM, and other similar models on the Reddit text dataset, allowing it to learn intricate patterns and relationships within the text data. Upon comparing the results of the two approaches, we observed that the BERT and DistilBERT consistently outperformed the general machine learning algorithms and some neural models like RNN and LSTM.**

***Index Terms*—NLP, SVM, Naive Bayes, Multinomial Naive Bayes, Stress, Confusion Matrix, BERT, SHAP, LIME, Attention**

I. Introduction

Modern life's rapid pace, constant connectivity, and work-related demands are fueling an increase in stress. The past pandemic further exacerbated stress through health concerns, remote work challenges, and disrupted routines. Working with stress is significant in light of the fact that stress has turned into an unavoidable and possibly crippling issue in present-day culture. The effects of weight on both individual prosperity and by and large general well-being are significant. Constant stress can prompt different physical and psychological well-being issues, going from cardiovascular illnesses to tension and sorrow. By creating models that can predict feelings of anxiety, we can distinguish those in danger and give opportune mediation, empowering people to embrace compelling survival techniques. Moreover, understanding the examples and triggers of stress through NLP can prompt experiences that illuminate fitted ways to deal with the stress of the executives, cultivating better ways of life and working on by and large personal satisfaction. Stress presents critical risks, obvious from insights showing its inescapable effect. Around 77% of people experience actual side effects because of stress, with 73% confronting mental impacts, according to the American Establishment of Stress. The American Mental Affiliation features that pressure costs the US economy more than $300 billion every year. Moreover, the World Wellbeing Association recognizes pressure as a main calculated psychological well-being issue, influencing 1 out of 4 individuals internationally. Using techniques like AI becomes basic to anticipate, forestall, and oversee pressure, relieving its destructive results on people and society. Stress and sentiment analysis share a connection in examining emotional expressions within textual data. By applying sentiment analysis to texts related to stress, such as social media texts or online discussions, it becomes possible to understand the emotional tone or feelings and sentiment associated with stress issues. This approach provides a structured method for understanding how people convey their stress-related feelings and allows for insights into common stressors, and potential triggers. Through sentiment analysis, researchers can derive valuable information to tailor interventions, develop support strategies, and gain a deeper comprehension of the emotional nuances surrounding them. Sentiment analysis is the technique of extracting subjective information from text. Sentiment analysis can assess the intensity of emotion in a text or speech. It is able to identify people who are stressed by examining their written messages, can create solutions for reducing stress, and determine the causes of stress, is possible to track variations in stress levels over time using sentiment analysis, help people control their stress, and customized stress treatments can be developed via sentiment analysis. Overall, sentiment analysis is an effective approach for locating, measuring, and monitoring stress levels. In addition, it can be implemented to create individualized stress-reduction programs. Sentiment analysis can be done using several neural network models. RNNs are used for sentiment analysis in order to identify long-term dependencies between texts. While LSTM requires greater computation, both GRU and LSTM networks have proven useful for sentiment analysis applications. To extract features from text, CNNs can also be used. CNNs can grab local patterns and features in text for sentiment analysis. An entire sentence's context is extracted by bidirectional LSTM networks. Sentiment analysis is improved by processes of attention and bidirectional LSTM models. Also, several transformer-based models can be used. BERT and GPT models are powerful models for sentiment analysis due to their bidirectional nature and contextual understanding. For sentiment analysis, attention models might concentrate on relevant words or phrases. For sentiment analysis, hierarchical models may represent both local and global contexts. The dataset and its use determine which neural network model is most effective for sentiment analysis. Experimentation is required to determine hyperparameters such as learning rate, number of layers, and neurons per layer. Before training the neural network, preprocessing the text data may improve performance.

II. Related Works

The article by Zhang et al. talks about how Natural language processing (NLP) has been used to identify mental illness using text over the past ten years [1]. NLP-driven psychological maladjustment location research is on the ascent, as per a story survey of 399 examinations distributed in the decade, and profound learning-based techniques have acquired prevalence lately.

The audit distinguished three principal sorts of datasets utilized for psychological sickness location: online entertainment posts, screening overviews, and clinical notes. Depression and suicide were the most frequently studied mental illnesses, followed by stress, anxiety, and schizophrenia.

The survey likewise found that an assortment of NLP techniques have been utilized for psychological sickness discovery, including conventional AI strategies, for example, support vector machines (SVMs) and gullible Bayes classifiers, as well as profound learning strategies like intermittent brain organizations (RNNs) and convolutional brain organizations (CNNs). Profound learning techniques have for the most part been displayed to beat conventional AI strategies.

According to the survey's reasoning, NLP may be an important tool for the early detection of mental illness. Notwithstanding, the creators brought up that there are as yet various obstructions that should be survived. A portion of these deterrents incorporates the shortfall of huge, top-notch datasets, the necessity for models that are straightforward, and the need to think about the moral ramifications of utilizing NLP to identify psychological instability.

In conclusion, NLP-driven psychological maladjustment location research has seen significant advancements in the past decade, with deep learning techniques leading the way. By leveraging textual data from online posts, screening surveys, and clinical notes, NLP has the potential to revolutionize mental health care by enabling early detection and personalized interventions. However, addressing challenges related to data quality, model transparency, and ethical considerations is essential for realizing the full potential of NLP in mental illness detection and ensuring its responsible and ethical use in healthcare settings.

The study emphasizes the significance of using cutting-edge, cost-effective methods like natural language processing (NLP) [2] for data collection and analysis in suicide prevention efforts. By joining NLP with existing information from Electronic Well-being Records (EHRs) and different libraries, specialists can upgrade the distinguishing proof and mediation processes for patients with a high probability of self-destruction endeavors. Additionally, unstructured clinician notes and other textual data presents opportunities to enhance the identification of various health conditions, mental health issues that resist treatment, and negative health outcomes are the mention able appliance natural language processing (NLP).

Notwithstanding, the exploration recognizes that moral contemplations, including security and information insurance, should be painstakingly tended to while using NLP for psychological wellness appraisal. Guaranteeing the mindful utilization of these advances and dealing with delicate information properly is critical to acquiring the trust of the two patients and clinicians.

The presented research by Qasim et al. [3] delves into the domain of text classification and explores the potential of transfer learning models, specifically BERT-based models, for automating text classification tasks. The study focuses on three diverse datasets related to COVID-19, covering fake news, English tweets, and extremist/non-extremist content. The objective is to assess the performance of transfer learning models on these datasets.

The research employs various advanced transfer learning techniques, including BERT-Base, BERT Large, RoBERTa-Base, RoBERTa-Large, DistilBERT, ALBERT-Base-v2, XLM-RoBERTa-Base, Electra-Small, and BART-Large [4] [5] [6] [7].

The datasets used in the study vary in size, with the COVID-19 fake news dataset containing over 10,000 instances, the COVID-19 English tweet dataset comprising almost 7,000 instances, and the extremist & non-extremist dataset being the largest with over 21,000 instances.

The methodology involves data preprocessing, encoding, model evaluation, and testing. After cleaning the datasets by removing URLs, converting text to lowercase, and lemmatization, the cleaned data is encoded and used to train and evaluate the transfer learning models. The evaluation metrics include accuracy, precision, recall, and F1-score. The results are compared against state-of-the-art techniques to highlight the fruitfulness of the proposed approach.

The experimental results show the impressive performance of the models on different datasets. For the COVID-19 fake news dataset, the RoBERTa-base model achieves a remarkable accuracy of 99.71%. On the COVID-19 English tweets dataset, the BART-large model emerges as the top performer with an accuracy of 98.83%. Finally, in the extremist & non-extremist datasets, both BERT-base and BERT-large models achieve an outstanding accuracy of 99.71%.

In conclusion, the research emphasizes the potential of transfer learning models, specifically BERT-based models, in text classification tasks. The findings highlight the exceptional performance of these models across various datasets. The researchers suggest future research directions, including the exploration of larger datasets, multiclass classification, and the incorporation of emoticons to enhance the accuracy and scope of text classification. The study underscores the power of transfer learning in automating text classification tasks and encourages further investigation into real-time sentiment analysis and broader social network analysis.

In a paper named Stress detection using Natural Language Processing and Machine Learning over social interactions [8], the authors Tanya Nijhawan, Girija Attigeri and T. Ananthakrishna worked on research of detecting stress using sentimental and emotional techniques on a large Twitter dataset that contains individual posts and comments. As a language model, they used BERT and an unsupervised machine-learning technique that adopted Latent Dirichlet Allocation. This unsupervised method helped to cluster similar genre data and types. LDA is used to calculate the topic structure from the density of the topic. Their deep learning-based BERT model classification helped to classify stress labeling. From this approach, they achieve around 94% of accuracy. They also compared with typical machine learning models but they found the BERT technique more evident. In their future work, they are willing to work on bi-polar sentiments and like to apply distinct platform datasets to justify the approach.
In paper by Kuma et al. mainly focuses on classifier techniques [9]. They went through a comparison and review analysis of multiple classifiers and justified which one gives a better outcome. The implementation went on a real-life dataset extracted from Twitter and evaluated the sentimental analysis on it. They used bigram, trigram, and unigram approaches with TF-IDF embedding system and applied through multiple classifiers. They found satisfactory accuracy from XGBoost and SVM models consecutively at 81% and 89%. Comparatively higher accuracy was found from a combination of LDA+Bigram+TF-IDF in the SVM model [10]. In their paper, they mentioned their future works on identifying the personalities from data on linguistic style.

Social media is an effective way to detect psychiatric problems, thereby preventing undiagnosed mental disorders. This paper [11], categorized postings from Reddit to determine stressful and non-stressful using machine learning models and multiple embedding methods. A high F1 score, Precision and Recall score of 0.76, 0.71, and 0.74 were obtained for the results. These results can be used to evaluate mental stress among social media users in real-world situations.

The ability to identify and provide support for difficulties with both physical and mental wellness depends increasingly on stress evaluation and sentiment analysis of posts on various websites of micro-blogging. To aid in early treatment for depressive disorders, this research provides an understanding of the issues and traits that stressed people faces across the world.

The Reddit's, a popular social media platform,posts from this platform have been taken into the consideration throughout this paper. This dataset contains 2800 texts for training. Reddit receives about 470,000 comments a single day, and there are 500 million tweets every day. People can express their emotions on social media platforms, and the volume of messages sent via text makes it easier to recognize signs of mental stress. To identify stress on social media, embedding methods and machine learning models are applied.

The following paper used the dataset of Dreaddit to retrieve mental stress from posts of Reddit. In order to categorize stressful postings on Reddit, NLP, and machine learning models like BERT, TF-IDF, and Word2Vec are utilized. The Dreaddit dataset, containing important metadata and text content, can be useful for NLP exercises, social media activity analysis, and stress analysis. The dataset is a text corpus containing the body, community name, label field, and other fields from a Reddit post. Pre-processing entails tokenization and text cleanup using NLP methods. For smaller datasets, conventional machine learning techniques are used.

This paper used the following methodologies: Removal of noise to improve text classification, Keywords combined to facilitate BERT and ELMo tokenization, ELMo and BERT used to create word embeddings and tokenization, ELMo model trained on cleaned text to form vectors as word embeddings, Machine learning algorithms trained on vectors to compare results.

RAKE (Rapid Automatic Keyword Extraction) is an algorithm that uses stoplists and scoring systems to extract relevant phrases and words from a target text. Elmo vectors generate an array of forms with 1024 parameters from contextualized encodings of word strings in each post. ML models are optimized by the use of multiple methods to extract features. In comparison to other test classifiers, LR, SVM, and XGBoost provide better results, with Logistic Regression having the greatest F1 score. The highest F1 score is produced using BoW (bag-of-words) embeddings with weighted TF-IDF vectors.

The foundational mathematics and stochastic nature of each method determine the accuracy, precision, and recall of a model. While the LR model employs hyperparameters to change performance measures, XGBoost builds new models using decision trees and other mathematical methods. The best F1 score that XGBoost has received is 0.70. SVMs use hyperplanes to differentiate between dataset classes while LSTM and BERT models extract distinctive characteristics to boost accuracy, giving them a combined F1 score of 0.76. The best results were from LR and SVM, with ELMo vectors and BERT embeddings outperforming the Bag-of-Words model in terms of performance.

Authors' mental states can be better understood by examining posts they have published across different platforms. Using PRAW API data, this paper successfully acknowledged stressful posts and produced significant results using conventional criteria for assessment. Important NLP research areas include sentiment analysis, text categorization, opinion mining, Word2vec, GloVe, and tf-idf vectors, which can enhance translation, author identification, spam detection, and language identification. It is possible to perform multimodal sentiment analysis by fusing text and visuals from different sources.

This study proposes using machine learning and natural language processing (NLP) to identify the indicators of human stress in diffrent social media platform posts. Using innovative techniques like ELMo embeddings along with established models like SVM can enhance the analysis of mental stress in social media. Future investigations in the realm of neural network-based models and models that are pre-trained can further expand upon the discoveries made in this study.Finally, this proposed method has the potential to identify symptoms and offer assistance to lessen mental health problems.

In this research paper by Turcan et al., the authors address the prevalent issue of stress in the online world, particularly in social media platforms [12]. Stress is a nearly universal human experience, and while it can serve as a motivator, excessive stress has been associated with negative health outcomes. Recognizing the importance of stress identification across various domains, the authors introduce the Dreaddit dataset, a comprehensive text corpus comprising over 190,000 posts from five distinct categories of Reddit communities.

The dataset's objective is to facilitate stress detection and analysis, with potential applications in fields such as diagnosing physical and mental illnesses, monitoring public sentiments in politics and economics, and assessing the impact of disasters. Unlike existing computational research that predominantly focuses on stress analysis in speech or short texts like Twitter, Dreaddit offers lengthy multi-domain social media data, allowing for a more comprehensive understanding of stress expression.

The research team carefully selects ten subreddits from five domains, covering topics such as interpersonal conflict, mental illness (anxiety and PTSD), and financial need. These subreddits serve as platforms for users to share personal experiences, seek advice, and provide support related to stressful situations. In the dataset 420 tokens is the average post length , which is significantly longer than any microblog data, enabling an in-depth examination of stress expressions.

To train supervised models for stress detection, the authors label 3,553 segments from the dataset using human annotators recruited via Amazon Mechanical Turk. The labeled data ensures a balanced representation of stressful and non-stressful content. Around 52.3% of the data are labeled as stressful class. Furthermore, the research team conducts a thorough data analysis, examining vocabulary patterns, lexical diversity, and syntactic complexity in each domain. The analysis reveals significant distinctions among domains, influencing the classification system's performance.

The authors experiment with various supervised models, including decision trees, Support Vector Machines (SVMs), Naive Bayes, Perceptron, and logistic regression. Input representations such as bag-of-n-grams, pre-trained Word2Vec embeddings, Word2Vec embeddings trained on the dataset, and BERT embeddings are tested to identify the most effective model. Additionally, neural models like bidirectional Gated Recurrent Neural Network (GRNN) and Convolutional Neural Network (CNN) are trained and compared to traditional models and BERT.

The best-performing model is a logistic regression classifier utilizing domain-specific Word2Vec embeddings, high-correlation features, and high-agreement data. This model achieves an impressive F1-score of 79.80 on the test set, comparable to the state-of-the-art BERT-based model. Although neural models demonstrate potential, their performance is hindered by the relatively small dataset size.

An error analysis of the models reveals that both tend to overclassify stress, particularly with low-agreement data and less explicit stress expressions. To further enhance stress detection accuracy, future work will focus on incorporating the context and intentions of the writers.

In conclusion, the Dreaddit dataset provides valuable insights into stress expressions in social media. The research's significant contributions lie in the development of accurate supervised models and the provision of a resourceful dataset for further research in stress analysis. The findings emphasize the importance of domain knowledge and lexical features in stress detection and the potential for utilizing large unlabeled datasets in neural models to improve performance.

III. Methodology

In our dataset, we are focusing on the text part, which contains stress and non-stress texts. As the target or prediction result, we took the label column. The label column contains either stress text or non-stress text. Our motive is to focus on a comparison analysis between classical machine learning models and deep learning models. At the very beginning, all features except text and label columns were dropped. As the text is natural, it is not well formatted. The texts contain HTML tags, punctuation, different symbols, and stopwords. These things do not carry any meaning in the sentences. Even they do not have that much effect on stress and non-stress classification. So after normalization and removing unnecessary symbols, we turned the text into a straightforward text that contains only words. To get the root forms of words, stemmers were used. These things were done in the preprocessing phase. To incorporate this data into the model, we must vectorize the words so that they can be understood. There are several word embedding options, including word2vec, TF-IDF, countervectorizer, and rapid text, among others. We attempted to use the word2vec and countervectorizer tools. Due to the fact that word2vector converts vectors to negative values, a few models were also challenging to adapt. Consequently, the word2vec embedding did not produce a great deal of satisfactory outcomes. As a result, a countvectorizer was chosen for the word embedding assignments to facilitate further comparison.

To start our study, we carefully chose several traditional machine learning models known for their distinct abilities. The initial models we picked were K-nearest neighbors (KNN), Support Vector Machines (SVMs), Multinomial Naïve Bayes, Random Forest classifier, Multi-layer Perceptron, Decision Tree classifier model, Logistic regression, XGBoost and Adaboost. This selection was intended to enable a solid comparison and analysis at a fundamental level, helping us uncover the complexities of the data [13] [10].

We had multiple reasons for choosing these non-neural models. Firstly, they represent a wide range of machine learning techniques, covering both linear and non-linear methods. This allowed us to explore different ways of modeling stress prediction, giving us a thorough perspective of the problem. Additionally, these models are well-established in the machine learning field and have a successful track record across various tasks, making them suitable candidates for our initial investigation.

Our aim in using these models was two-fold. Initially, we expected these models to provide useful baseline results that could act as reference points for assessing more advanced models. Secondly, we aimed to identify any inherent patterns or tendencies in the data that could guide us toward the most effective modeling approach.

While the initial non-neural models provided valuable insights, we saw room for improvement. One notable area for enhancement was incorporating neural models, specifically those employing attention mechanisms. Neural models, particularly those with attention mechanisms, have shown a remarkable ability to capture complex patterns and relationships in intricate datasets. The unique feature of attention mechanisms, allowing the model to focus on specific parts of the input data, could be crucial in identifying subtle cues related to human stress levels.

Introducing neural models with attention mechanisms potentially enhances our stress prediction performance. The adaptable nature of attention allows the model to dynamically assign importance to different features, helping it recognize and prioritize relevant information. This aligns well with the complex nature of human stress, which often involves intricate interactions between various factors. Moreover, neural models can learn hierarchical representations, potentially revealing hidden features that conventional machine learning models might miss.

Additionally, the use of pre-trained GloVe embeddings to align the dataset meticulously within the framework of deep learning models. The architecture utilized for the Recurrent Neural Network (RNN) was the Long Short-Term Memory (LSTM), which also included Convolutional Neural Network (CNN) elements in the LSTM model [14]. The training approach for each of these deep learning models consisted of 35 epochs, with a batch size of 45.

The development of this strategic approach was influenced by the understanding that LSTM neurons have the ability to comprehend complex contextual details, allowing them to effectively distinguish between important and unimportant phrases. The inherent capacity shown here is crucial for accurately forecasting different types of stress by comprehending contextual subtleties.

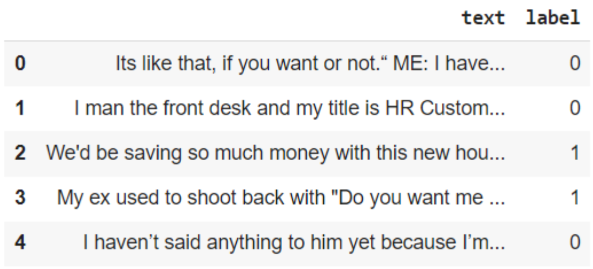
In the context of BERT and DistilBERT, both models share similar methodologies in comparison to the previous models. However, the process of encoding input texts differs. The default tokenization techniques provided by the transformers library and TensorFlow Hub modules are employed to preprocess and tokenize the training samples prior to feeding them into these models. This involves breaking down the text into smaller units, such as words or subwords, and adding special tokens like [CLS] and [SEP] for classification and separation. This tokenization process helps preserve the models' remarkable context-capturing capabilities, as it enables them to understand the relationships between words and their positions in the text, capturing intricate contextual nuances essential for tasks like natural language understanding and sentiment analysis. The number of epochs used for the BERT model is 70 with its default batch size, whereas the number of epochs used for the DistilBERT model is 20 with a batch size of 8. This allows BERT and DistilBERT to create rich, contextualized representations of words and sentences, resulting in enhanced performance across a wide range of natural language processing tasks.

In summary, our research paper conducts a comparative study of predicting human stress. We begin by examining various traditional non-neural machine learning models, which provide a solid foundation and prepare the groundwork for introducing neural models with attention mechanisms, Recurrent Neural Networks (RNN) with Long Short-Term Memory (LSTM), and Convolutional Neural Networks (CNN) with LSTM. This approach is expected to lead to improved performance and a deeper understanding of the factors influencing human stress levels.

IV. Data Analysis

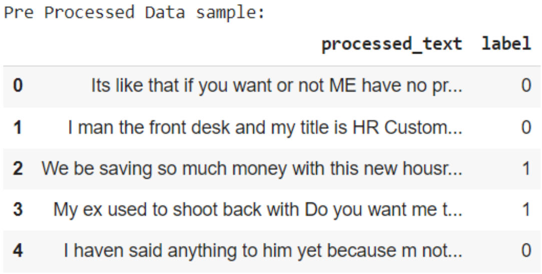
As our motive is to work with stress and not stress text classification, so we chose a dataset from Kaggle that is scrapped and labeled. It contained texts from social media from different users, and according to their texts stress and not stress is defined. This dataset was also used on a published paper, which inspired us to explore it again. The dimension of the dataset is (3553, 116). We chose all the rows and two columns for our project. The rest of the columns were dropped during pre-processing.

Before doing preprocessing, the duplicate values were checked. From the whole dataset, there were no duplicates in the text column. So all our 3553 values were unique and balanced. After the duplication test, our text column went through a preprocessing phase. As it is real-world data, it has many unnecessary symbols, characters, and marks that do not carry any meaning. Those were removed during the preprocessing phase. Besides, there are multiple forms of words, which makes for a scattering of words. To unite them to the root form of words, stemming was done. (i.e. looking → look, looks → look etc)



	text	label
0	Its like that, if you want or not.“ ME: I have...	0
1	I man the front desk and my title is HR Custom...	0
2	We'd be saving so much money with this new hou...	1
3	My ex used to shoot back with "Do you want me ...	1
4	I haven’t said anything to him yet because I’m...	0

Fig. 1. Before Preprocessing



Pre Processed Data sample:

	processed_text	label
0	Its like that if you want or not ME have no pr...	0
1	I man the front desk and my title is HR Custom...	0
2	We be saving so much money with this new housr...	1
3	My ex used to shoot back with Do you want me t...	1
4	I haven said anything to him yet because m not...	0

Fig. 2. After Preprocessing

After doing so we split the training and testing data in an 80:20 ratio. In total, we checked the stress and not stress total data were almost equal. Though the not-stress data was 200 more. As the data split was randomly shuffled it probably won’t affect that much to the training phase much.

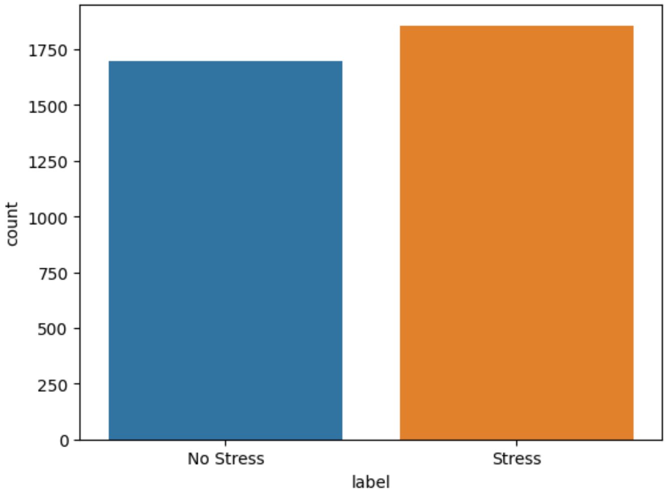


Fig. 3. Stress and not stress ratio

A. WordCloud

WordCloud is a visualization tool by which we can picture words according to their importance or frequency. From our first sample, we can see the word ‘feel’ is used mostly in the context. As the dataset was taken from “Stress Analysis in Social Media”, we can understand that the users mostly used words like ‘feel’,‘know’,’time’,’even’,’want’,’really’,’help’. The size of the words is the induction of their frequency. More the words are found in the corpus, the larger they appear in the word cloud. This word cloud was generated Based on Stress Words.

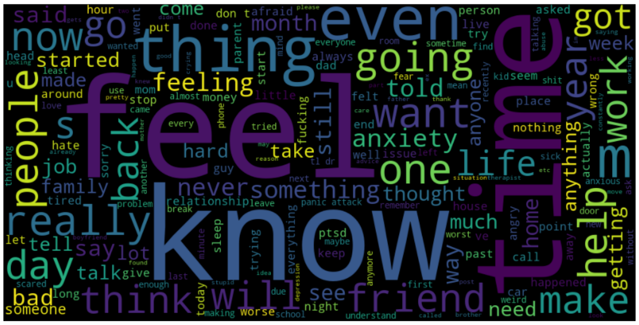


Fig. 4. Word Cloud Based on Stress Words

If we generate the word Cloud Based on Not Stress Words of the corpus the word sizes and colors look different. Sentiment-wise, the word ’know’ gets the most highlights in this corpus After that, rest of the words like ’want’, ‘feel’, and ‘time’ -words in the word cloud indicate the need of good classification models as we can see overlaps with the stress words. This is an effective content prioritization. From this, we can tell which words are associated with the sentiment of the users.

want
feel
time
know
one
really
people
much
now
help
back
need
go
year
thing
good
think
going
make
work
friend
even
will
way
life
never
day

Fig. 5. Word Cloud Base on Base on Not Stress Words

The following pie chart shows the cause of people's stress. From the chart, we can see that most people's stress is caused by PTSD. Around 20% of people are having stress because of PTSD. In second place, approximately, 19% of the people are stressing over relationship problems. Another leading cause of people's stress is anxiety. Anxiety is responsible for around 17% of people's trees which makes it the third leading cause of stress. Besides these three, we can observe that there are other factors that are causing stress in people's lives. Some such causes are domestic violence, assistance, survivors of abuse, etc. We can visualize the ratio of these factors with respect to the three leading factors from our pie chart. In short, this pie chart presents a comparison between the factors of stress and gives insight into the leading causes of stress.

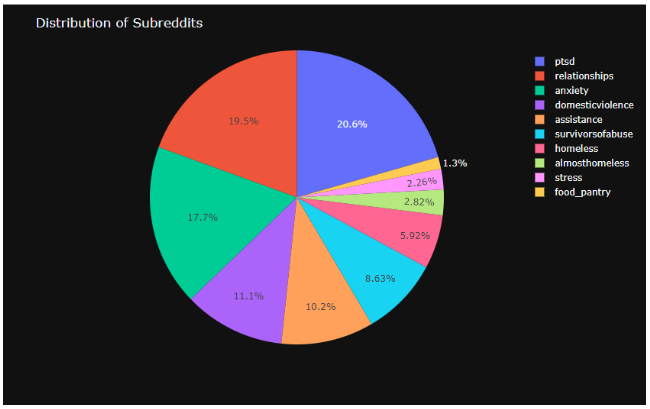


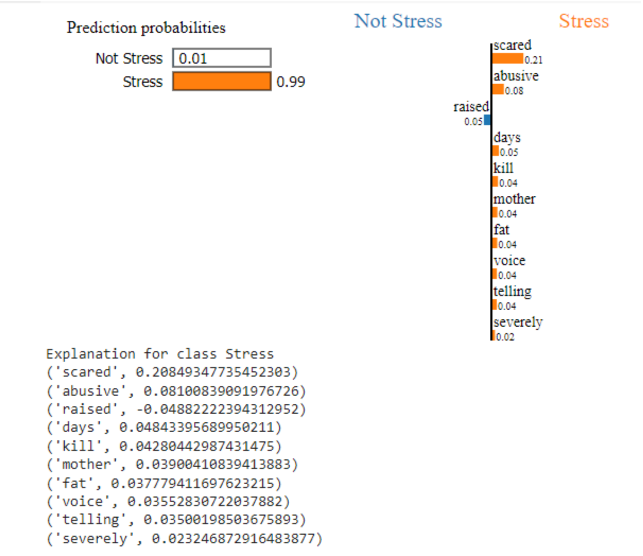
Fig. 6. Distribution of stress types

V. Explanation with XAI

XAI means Explainable Artificial Intelligence which refers to the some techniques and approaches with the goal to make the decisions and processes of AI systems understandable and transparent to humans. The need for these technics arises as many AI models are often seen as "black boxes" which meaning their inner workings are complex and not easily interpretable by human . Deep learning models BERT and attention based models are such type of models. As we used this types of model in our task, we will use LIME aka Local Interpretable Model-agnostic Explanations and SHAP also know as Shapley Additive Explaination techniques which belongs to the Explainable Artificial Intelligence (XAI) [15] [16]. They help make black-box machine learning models more transparent by providing explanations and global feature importance.

A. Using Lime

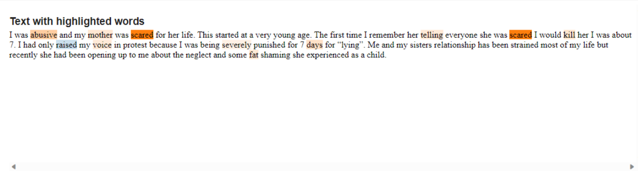
Our Lime generated output shows that 99% of our given post indicates stress whereas only 1% shows there is no stress in the post corpus.In this particular post 'scared' has 21% feature importance and 'abusive' has 8% importance. These makes the model understand that there is stress press in this post.



```
Explanation for class Stress
('scared', 0.20849347735452303)
('abusive', 0.08100839091976726)
('raised', -0.04882222394312952)
('days', 0.04843395689950211)
('kill', 0.04280442987431475)
('mother', 0.03900410839413883)
('fat', 0.03777941169762321 5)
('voice', 0.03552830722037882)
('telling', 0.03500198503675893)
('severely', 0.023246872916483877)
```

Fig. 7. Lime Prediction result

For this Prediction we used a post from the dataset to calculate its LIME prediction.



Text with highlighted words

I was abusive and my mother was scared for her life. This started at a very young age. The first time I remember her telling everyone she was scared I would kill her I was about 7. I had only raised my voice in protest because I was being severely punished for 7 days for "lying". Me and my sisters relationship has been strained most of my life but recently she had been opening up to me about the neglect and some fat shaming she experienced as a child.

Fig. 8. The sentence used for Lime Prediction

B. Using Shap

" Have been working at a factory job for 1 month, hoping it would be a career change. Pretty physical job, push/pulling things that are up to 150lb quite often during my day. - Came into this job while recovering from a bulging disc in my lower back (L5/S1) - A medical certificate was provided to my employer advising that I have a back injury going into this job. ” For this post the SHAP word analysis we found is shown below.

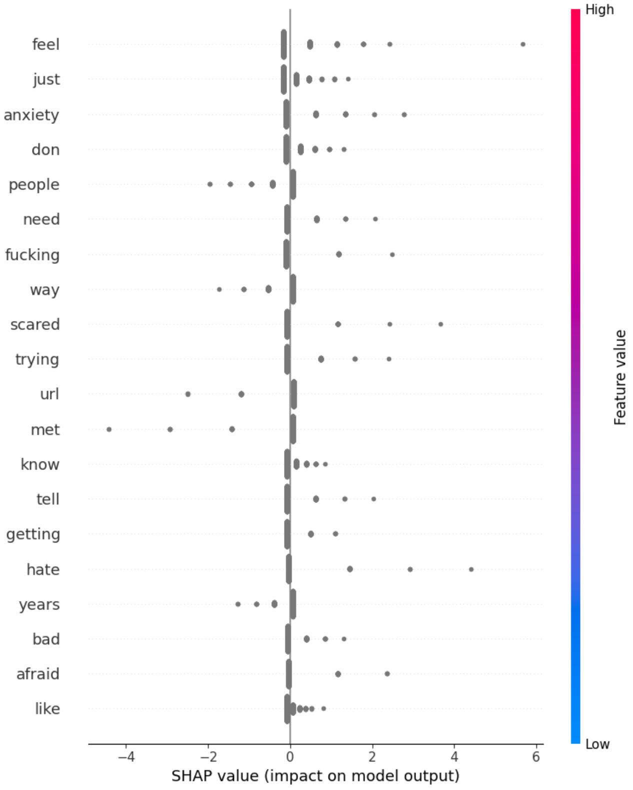


Fig. 9. Overall SHAP word analysis in sentence

From this Beeswarm plot by SHAP, we can see that word such as ’feel’,’juts’,’anxiety’ have more fature value than ’people’,’met’ and ’url’. Which leads us to the point where we can understand that words with higher SHAP value indicates that these words are stressed word. Present of such words make the models predict strees in any text corpus.This way using LIME and SHAP analysis human get the gist how these balck box model works.

VI. Result Analysis

In this section we will discuss the performance of all the models we have used. In total we used 14 model which are catagorized in two types of model such as neural and non-neural models. The results we have got from our models, we evaluated them and compare them using their confuaion matrix. All the confusion Matrix of each model are shown below.

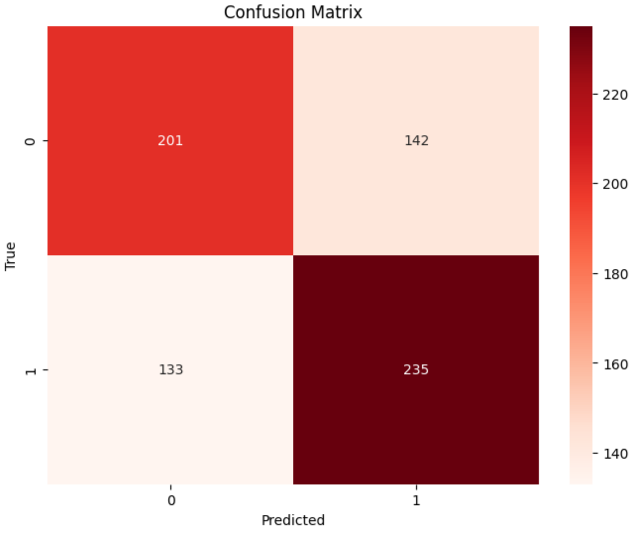


Fig. 10. Confusion Matrix report for the Decision Tree classifier model

To forecast the class or value of each data point, decision trees create a tree-like structure of decisions. The decision tree achieved an accuracy rate of 61.32%. The decision tree model was effective in predicting ”Stress” and ”Not-Stress” instances, with an F1-Score of 63.09% and 59.38%, respectively.

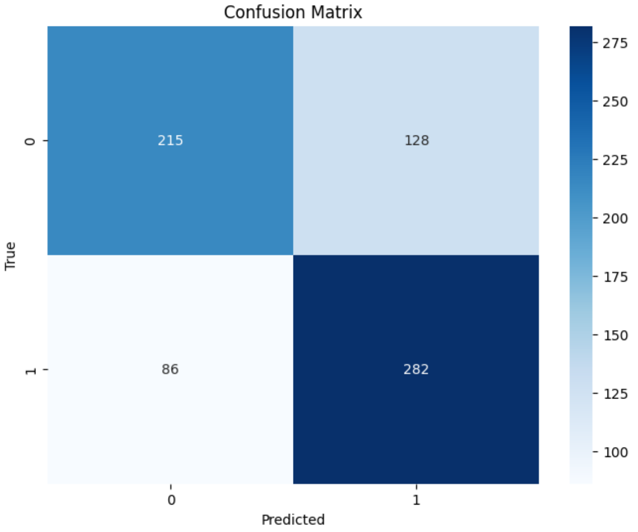


Fig. 11. Confusion Matrix report for the MLP classifier model

Multilayer perceptrons (MLPs) are artificial neural networks that are used for classification and regression. The MLP classifier had an accuracy rate of 69.90% and an F1-Score of 72.49% for identifying ”Stress” cases and 66.77% for identifying ”Not-Stress” cases. The MLP classifier performed well in both stressful and not-stressed circumstances.

Another reputed classifier we used is the AdaBoost classifier. The AdaBoost classifier showed an overall accuracy of approximately 67.4%. The model classified the ”No Stress” class with a precision of 66%. For the ”Stress” class, it was 68%, meaning that 68% of instances predicted as ”Stress” were indeed correct. We got 66% and 69% as the recall from this model for the ”No Stress” and ”Stress” class. This model provides an F1-score of 0.66 for the ”No Stress” class, and 0.69 for the ”Stress” class. In short, this model performed better than many models but yet MLP classifier works better for this particular task.

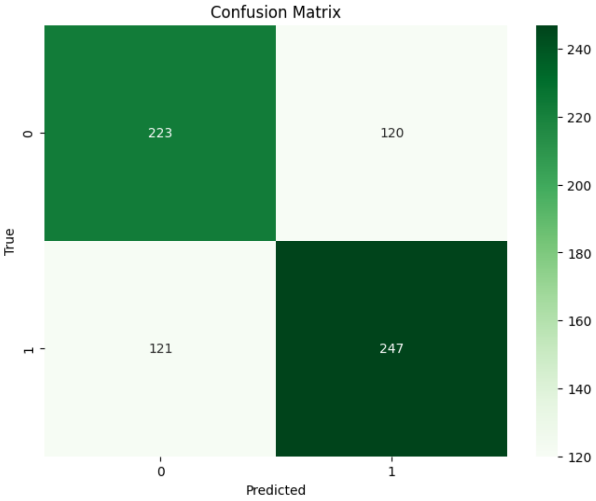


Fig. 12. Confusion Matrix of the AdaBoost classifier model

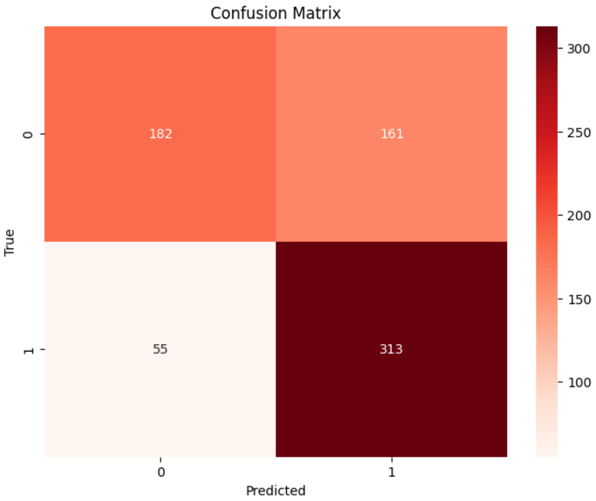


Fig. 13. Confusion Matrix report for the K-nearest neighbors (KNN)

We use another non-parametric, supervised learning algorithm K-nearest neighbors (KNN). The fundamental idea behind k-NN is that similar data points tend to have similar outcomes. It operates on the principle that if most of the nearest neighbors of a data point belong to a certain class, then that data point is likely to belong to the same class. The model attained an overall accuracy of 0.5907. It demonstrated an F1-Score of 0.5689 in ”Stress” cases and also achieved F1-Score 0.6104 in identifying ”Not-Stress” Cases.

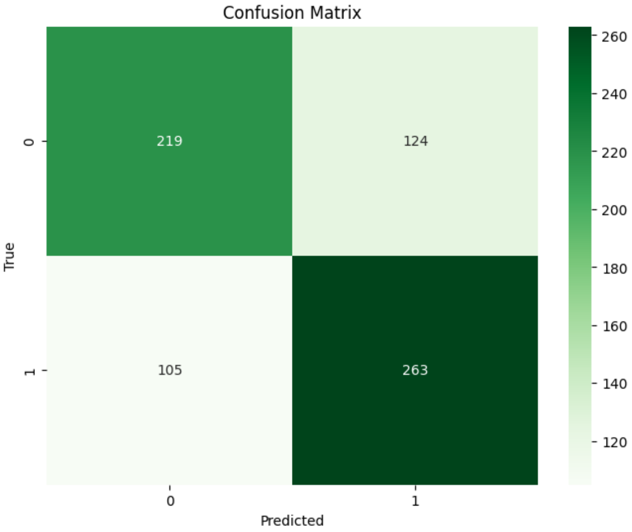


Fig. 14. Confusion Matrix report for the SVM classifier model

We use a supervised machine learning algorithm named Support Vector Machine(SVM). Support Vector Machine (SVM) is a versatile machine learning algorithm primarily used for classification and regression tasks. It is particularly well-suited for scenarios where the data is not linearly separable or has a complex decision boundary. The overall accuracy of this model is 0.6779. It has an F1-Score of 0.6967 for identifying ”Stress” and 0.6567 for identifying ”Not-Stress”.

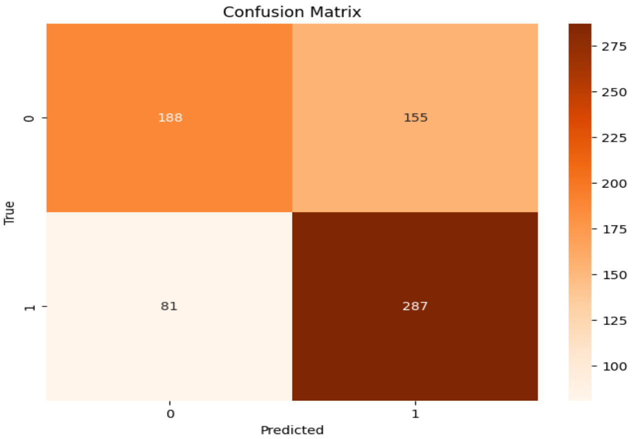


Fig. 15. Confusion Matrix report for the Random Forest

Random Forest is a robust method for machine learning that can be used for classification, regression, and clustering. The Random Forest model has a 66.81% accuracy. This means that the model properly forecasts whether a person is stressed or not 66.81% of the time. The F1-score for classifying ”Stress” instances is 70.86%, whereas the F1-score for classifying ”Not-Stress” instances is 61.44%.

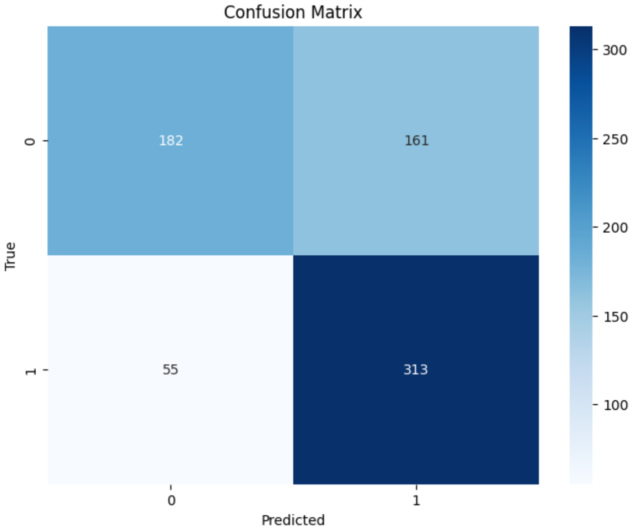


Fig. 16. Confusion Matrix report for the Naive Bayes model

Naive Bayes is a popular machine-learning algorithm used for classification tasks. It is called naive because it assumes that all features are independent of each other. Multinomial Naive Bayes is a special variant that is suited for text classification tasks. Our implementation of this model shows an overall accuracy of 0.6962. The algorithm shows an F1-score of 0.7435 classifying ”Stress”.

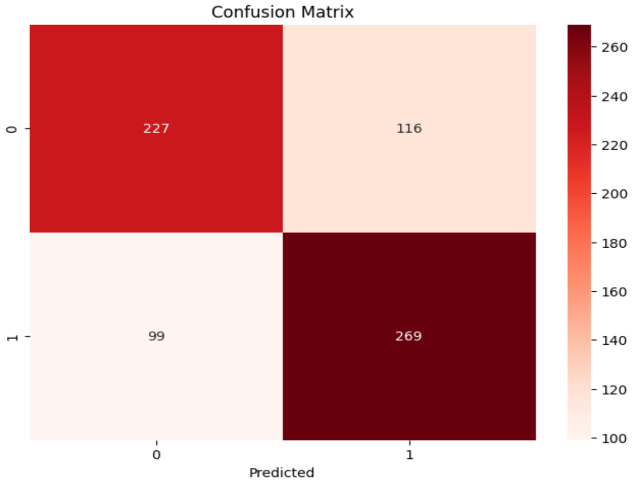


Fig. 17. Confusion Matrix of the Logistic Regression model

Moreover, the Logistic Regression statistical model was implemented. This model, like other machine learning models, produced an average score. Such as an accuracy of 69.76%, F-1 was 71.45% for texts with stress and 67.86% for sentences without stress.

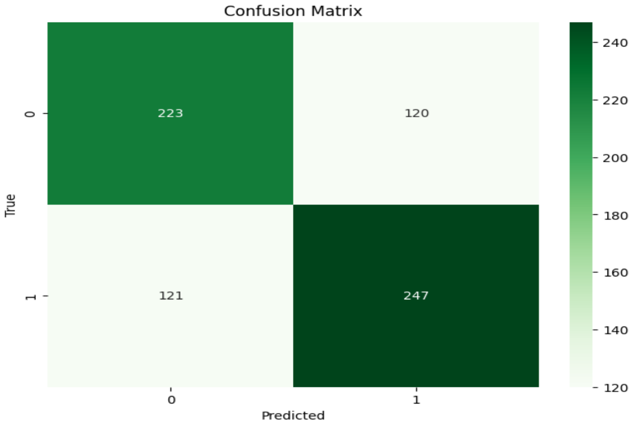


Fig. 18. Confusion Matrix of the AdaBoost model

The Adaboost algorithm was also applied to our dataset. The efficacy of this model was comparable to that of its competitors. The attained results were accurate to the extent of 66.10 percent. In addition, the F-1 score for stress-related texts was 67.21 percent, while the score for non-stress sentences was 64.92 percent.

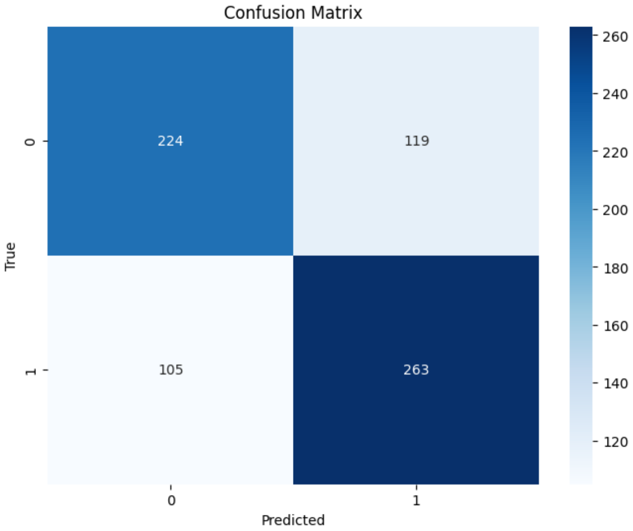


Fig. 19. Confusion Matrix of the XGBoost model

Most of the time, XGBoost provides promising results for NLP tasks. Consequently, we compared this model to other used models. In XGBoost, we achieved 68.50% accuracy. In addition, the F-1 score for texts with stress was 70.13 percent, while the score for texts without stress was 66.6 percent. Also in this case, we did not achieve the expectedly superior results compared to other machine learning models.

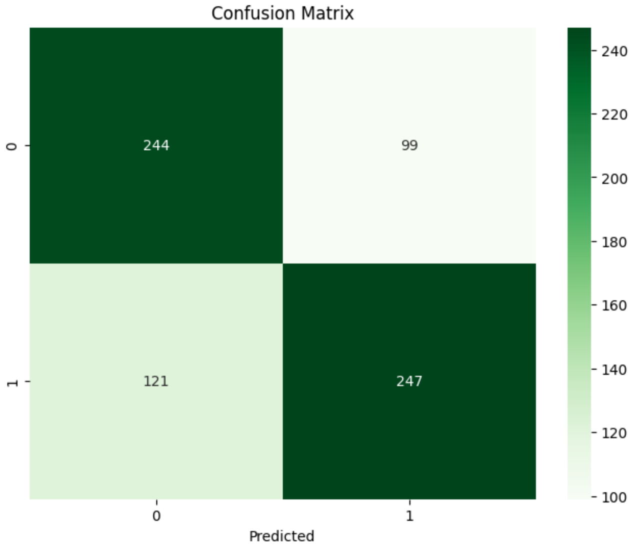


Fig. 20. Confusion Matrix of the Attention-LSTM model

After That we used some Attention-based Models for our task. First we worked with Attention-LSTM. This model was able to perform with the overall accuracy of 69.06%. We Achieved an F1-Score of 69.19% for identifying cases of ”Stress”. And this model demonstrated an F1-Score of 68.93

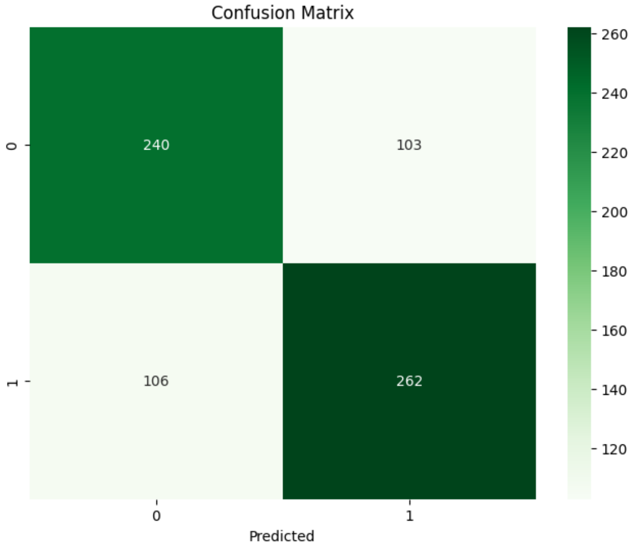


Fig. 21. Confusion Matrix of the RNN-LSTM model

Our Next model performed even better than Attention-LSTM. Recurrent Neural Network - LSTM (RNN-LSTM) performed with an accuracy rate of 70.60%. This model demonstrated an F1-Score of 71.49% for identifying instances of ”Stress” and an F1-Score of 69.67% for identifying instances of ”Not-Stress”.

Convolutional Neural Network - LSTM (CNN-LSTM) performed better than RNN-LSTM with an accuracy of 72.15%. We were able to achieve an F1-Score of 73.95% and 70.09% for ”Stress”, ”Not-Stress” class.

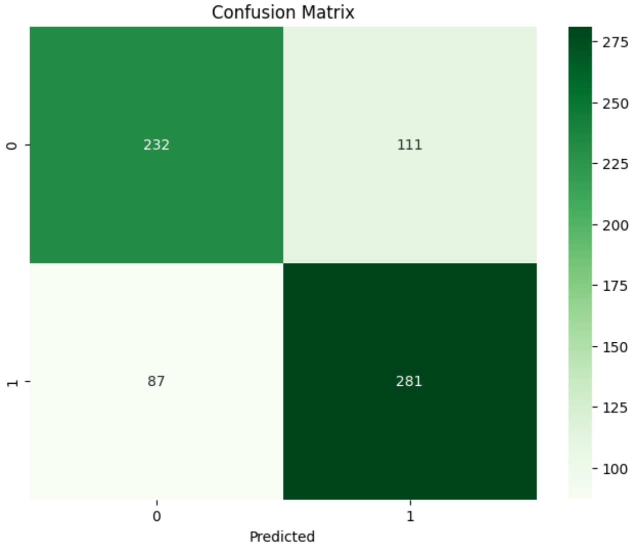


Fig. 22. Confusion Matrix of the CNN-LSTM model

We implemented the BERT model as the next model. It demonstrated a great performance achieving the second-best accuracy rate of 76.79%. Of particular significance is its

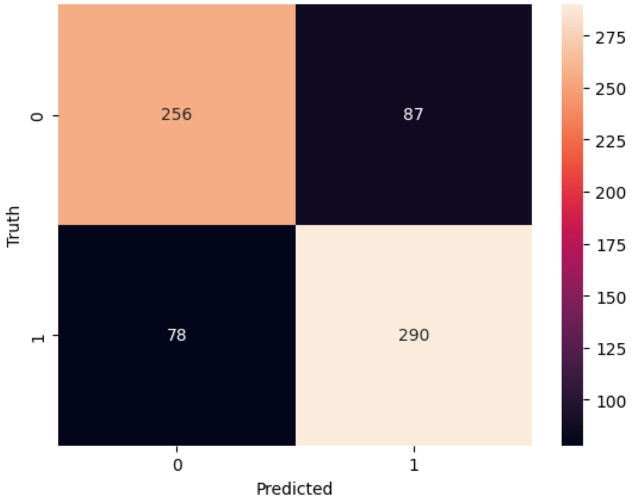


Fig. 23. Confusion Matrix of the Bert model

remarkable proficiency in correctly identifying instances categorized as ”Stress,” evident in its impressive F1-Score of 78.00% achieved for such cases. Additionally, BERT exhibited a commendable F1-Score of 76.00% in the accurate classification of cases falling under the ”Not-Stress” category.

Lastly, our next implementation was the DistilBERT model, which is the top-performing model from all of our tested ones in terms of accuracy. Impressively, DistilBERT attained a compelling accuracy rate of 77.91%. Particularly striking is its efficacy in recognizing instances of ”Stress,” as underscored by the exceptional F1-Score of 80.00% achieved for this specific classification. Furthermore, DistilBERT demonstrated a consistent F1-Score of 76.00% in labeling texts that fall in the ”Not-Stress” category.

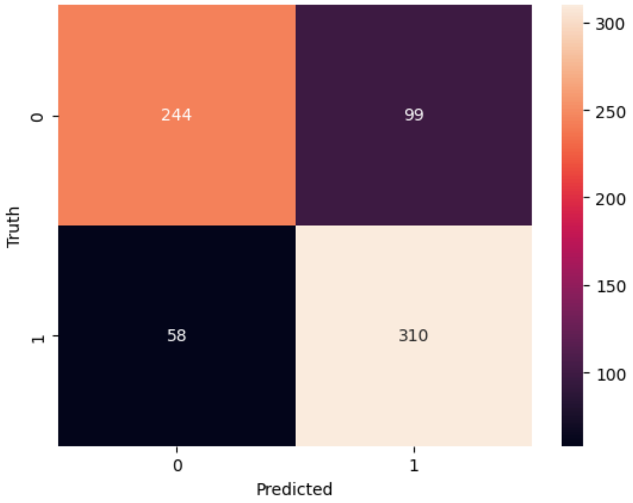


Fig. 24. Confusion Matrix of the DistilBERT model

A. Comparative Discussion

The realm of Natural Language Processing (NLP) has witnessed remarkable advancements owing to the rapid evolution of both traditional machine learning algorithms and sophisticated deep learning techniques. This comprehensive comparative analysis delves into the intricate performance details of a diverse array of models employed for the task of stress classification. By evaluating their accuracies and F1 scores across stress and non-stress classes, we aim to provide a comprehensive understanding of the contrast between neural-based models and conventional machine learning algorithms and their respective efficacies in stress classification.

1) Traditional Machine Learning Models: The examination of traditional machine learning models reveals notable insights into their performance in stress classification. The Multinomial Naive Bayes (MNB) model records an accuracy of 69.62%, with F1-scores of 74.35% and 62.76% for stress and non-stress categories, respectively. The Support Vector Classifier (SVC) yields an accuracy of 67.79%, accompanied by F1-scores of 69.67% and 65.67% for stress and non-stress labels. The k-Nearest Neighbors (KNN) model attains an accuracy of 59.07% while showcasing F1-scores of 56.89% and 61.04% for stress and non-stress classifications, respectively. The Random Forest model demonstrates an accuracy of 66.81%, and its F1-scores amount to 70.8% for stress and 61.44% for non-stress. Similarly, the Decision Tree model’s accuracy stands at 61.32%, with F1 scores of 63.09% and 59.38% for stress and non-stress categories. The AdaBoost model achieves an accuracy of 66.10% and F1-scores of 67.21% and 64.92% for stress and non-stress classes, respectively. Lastly, the Logistic Regression model secures an accuracy of 69.76%, accompanied by F1-scores of 71.45% and 67.86% for stress and non-stress labels.

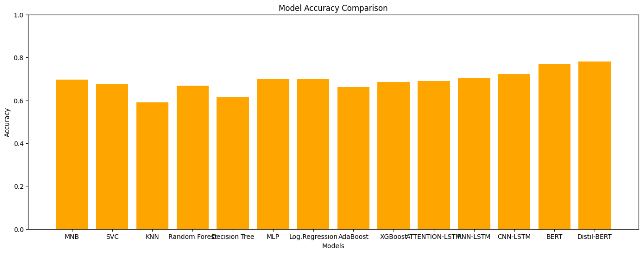


Fig. 25. Accuracy Comparison

Traditional models offer simplicity and interpretability, making them suitable for scenarios where model transparency is essential. However, they may struggle to encapsulate intricate language patterns, constraining their efficacy in tasks requiring nuanced understanding.

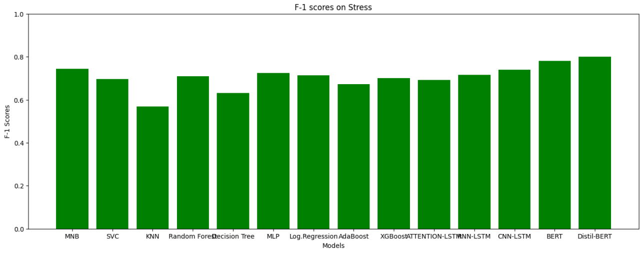


Fig. 26. F1-Score comparison for Stress

2) Deep Learning Models: In contrast, deep learning models exhibit distinctly superior performance in stress classification, underscoring the potential of neural-based methodologies in NLP tasks. The Attention-LSTM model, with an accuracy of 69.06%, achieves F1-scores of 69.19% for stress and 68.93% for non-stress categories. The RNN-LSTM model surpasses this with an accuracy of 70.60% and F1-scores of 71.49% and 69.67% for stress and non-stress classifications. The CNN-LSTM model, displaying an accuracy of 72.15%, boasts

F1-scores of 73.95% for stress and 70.09% for non-stress categories.

The pinnacle of performance is achieved by the BERT (Bidirectional Encoder Representations from Transformers) model, recording an impressive accuracy of 76.79%. Its F1-scores of 78.00% and 76.00% for stress and non-stress labels demonstrate the remarkable proficiency of BERT in capturing intricate language nuances. Distil-BERT follows closely with an accuracy of 77.91%, and F1-scores of 80.00% and 76.00% for stress and non-stress classifications, reinforcing the potency of transformer-based architectures in understanding contextual information.

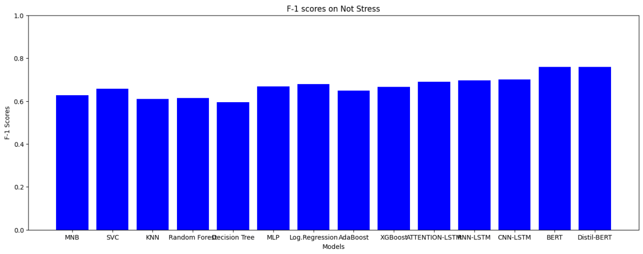


Fig. 27. F1-Score comparison for No Stress

3) Neural vs. Traditional Models: The substantial performance disparity between neural-based models and traditional machine learning algorithms underscores the supremacy of deep learning techniques in NLP, particularly stress classification. Neural models inherently possess the capacity to unravel intricate patterns and relationships in textual data, thus comprehending contextual subtleties that often elude conventional models. This observation is substantiated by the considerably higher F1-scores achieved by neural models, signifying their adeptness in both precision and recall for both stress and non-stress categories.

Moreover, deep learning models eliminate the need for manual feature engineering, as they can automatically extract pertinent features from raw text. This not only streamlines model development but also equips models to adapt more efficiently to diverse linguistic patterns.

4) Verdict: In summation, our meticulous comparative analysis underscores the preeminence of neural-based models, underpinned by the advancement of deep learning techniques, in outperforming traditional machine learning models in stress classification tasks. The findings accentuate the remarkable capability of these models to decipher intricate linguistic relationships and contextual intricacies within textual data. While traditional models hold their merit in scenarios where interpretability reigns supreme, neural models stand as the optimal choice for tasks necessitating an in-depth comprehension of language semantics. As the field of NLP continues to evolve, harnessing the potential of deep learning paradigms, epitomized by BERT and Distil-BERT, remains pivotal for attaining cutting-edge outcomes in stress classification and analogous language-centric undertakings.

TABLE I
RESULTS OF THE MODELS

Model Name	***Accuracy***	***F1-Score on Stress***	***F1-Score on Not-Stress***
MultinomialNB	69.62%	74.35%	62.76%
SVC	67.79%	69.67%	65.67%
KNN	59.07%	56.89%	61.04%
Random Forest Classifier	66.81%	70.86%	61.44%
Decision Tree Classifier	61.32%	63.09%	59.38%
MLP	69.90%	72.49%	66.77%
Log.Regression	69.76%	71.45%	67.86%
AdaBoost	66.10%	67.21%	64.92%
XGBoost	68.50%	70.13%	66.67%
ATTENTION-LSTM	69.06%	69.19%	68.93%
RNN-LSTM	70.60%	71.49%	69.67%
CNN-LSTM	72.15%	73.95%	70.09%
BERT	76.79%	78.00%	76.00%
Distil-BERT	**77.91%**	**80.00%**	**76.00%**

VII. CONCLUSION

In this work, we compared 14 machine-learning algorithms for human stress prediction. Among them, Deep learning and transformer-based models outperformed conventional machine learning models. Two transformer-based models: BERT and Distil-BERT are top performers with accuracy of 76.79% and 77.91%, respectively, while deep learning models: RNN and CNN also performed well. Although, there was not a significant amount of data in the datasets. The more data given, the better the models will learn to predict stress. There is a future scope of applying Generative adversarial networks (GANs) and Reinforcement learning(RL) to calibrate the result. Alongside textual data, voice data can be used. There can be a scope of detecting Stress Severity as predicting stress severity might assist people in receiving proper care.

REFERENCES

[1] T. Zhang, A. M. Schoene, S. Ji, and S. Ananiadou, "Natural language processing applied to mental illness detection: a narrative review," *NPJ digital medicine*, vol. 5, no. 1, p. 46, 2022.

[2] N. Indurkhya and F. J. Damerau, *Handbook of Natural Language Processing*, 2nd ed. Chapman & Hall/CRC, 2010.

[3] R. Qasim, W. H. Bangyal, M. A. Alqarni, and A. Ali Almazroi, "A fine-tuned bert-based transfer learning approach for text classification," *Journal of Healthcare Engineering*, vol. 2022, p. 3498123, Jan 2022. [Online]. Available: https://doi.org/10.1155/2022/3498123

[4] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *CoRR*, vol. abs/1810.04805, 2018. [Online]. Available: http://arxiv.org/abs/1810.04805

[5] K. Clark, M. Luong, Q. V. Le, and C. D. Manning, “ELECTRA: pre-training text encoders as discriminators rather than generators,” *CoRR*, vol. abs/2003.10555, 2020. [Online]. Available: https://arxiv.org/abs/2003.10555

[6] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, “Roberta: A robustly optimized BERT pretraining approach,” *CoRR*, vol. abs/1907.11692, 2019. [Online]. Available: http://arxiv.org/abs/1907.11692

[7] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, “Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter,” *CoRR*, vol. abs/1910.01108, 2019. [Online]. Available: http://arxiv.org/abs/1910.01108

[8] T. Nijhawan, G. Attigeri, and T. Ananthakrishna, “Stress detection using natural language processing and machine learning over social interactions,” *Journal of Big Data*, vol. 9, no. 1, pp. 1–24, 2022.

[9] P. Kumar, P. Samanta, S. Dutta, M. Chatterjee, and D. Sarkar, “Feature based depression detection from twitter data using machine learning techniques,” *Journal of Scientific Research*, vol. 66, no. 2, pp. 220–228, 2022.

[10] T. Chen and C. Guestrin, “Xgboost: A scalable tree boosting system,” *CoRR*, vol. abs/1603.02754, 2016. [Online]. Available: http://arxiv.org/abs/1603.02754

[11] S. Inamdar, R. Chapekar, S. Gite, and B. Pradhan, “Machine learning driven mental stress detection on reddit posts using natural language processing,” *Human-Centric Intelligent Systems*, vol. 3, no. 2, pp. 80–91, 2023.

[12] E. Turcan and K. McKeown, “Dreaddit: A Reddit dataset for stress analysis in social media,” in *Proceedings of the Tenth International Workshop on Health Text Mining and Information Analysis (LOUHI 2019)*. Hong Kong: Association for Computational Linguistics, Nov. 2019, pp. 97–107. [Online]. Available: https://aclanthology.org/D19-6213

[13] L. Breiman, “Random forests,” *Machine Learning*, vol. 45, no. 1, pp. 5–32, Oct 2001. [Online]. Available: https://doi.org/10.1023/A:1010933404324

[14] A. Sherstinsky, “Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network,” *CoRR*, vol. abs/1808.03314, 2018. [Online]. Available: http://arxiv.org/abs/1808.03314

[15] M. T. Ribeiro, S. Singh, and C. Guestrin, “”why should I trust you?”: Explaining the predictions of any classifier,” *CoRR*, vol. abs/1602.04938, 2016. [Online]. Available: http://arxiv.org/abs/1602.04938

[16] S. M. Lundberg and S. Lee, “A unified approach to interpreting model predictions,” *CoRR*, vol. abs/1705.07874, 2017. [Online]. Available: http://arxiv.org/abs/1705.07874