Stress Detection using Machine Learning

Mohsinali Vijapura*(AU2040080) , Manan Vadaliya*(AU2040264) ,Pruthviraj Dodiya*(AU2040175) , Mitsu Sojitra* (AU2040157)

School of Engineering and Applied Science, Ahmedabad University

***All Authors have contributed equally**

***Abstract*— The mental health of people is being threatened by stress, anxiety, and depression. Everybody's hectic lifestyle has a cause. Individuals frequently seek advice regarding their lives on subreddits of social media sites like Reddit and Instagram, where they publish updates and tales about their lives. Several content creators have stepped up in the last few years to provide information that will assist individuals with their mental health.**
In this project, we will perform EDA on the data to better understand the features, and then, using a variety of classification models, we will attempt to predict a person's level of stress based on the text that has been provided by the user.

***Keywords*— "data preprocessing," "feature extraction," "model training," "validation," and "performance metrics."**

I. Introduction

Stress is a pervasive issue that affects individuals of all ages, genders, and backgrounds. To address this issue, our team is developing a stress detection machine learning model to accurately identify signs of stress in individuals. We are using multiple sources of data to capture a more comprehensive picture of an individual's stress response. Our data analysis involves preprocessing and feature extraction, followed by machine learning algorithms for classification and prediction. Our motivation is to provide a reliable and efficient tool for stress detection, which can be integrated into various applications such as health monitoring systems, workplace wellness programs, and mental health support services.Our model shows promising performance in predicting stress levels, with an accuracy of over 80%. However, we have encountered challenges such as data quality issues, algorithmic complexity, and model interpretability. We are addressing these challenges by refining our data collection methods, optimizing our algorithms, and exploring different ways to visualize and interpret our model results. We plan to further validate and refine our model by collecting more diverse and representative data and comparing it with clinical assessments.

II. Literature Survey

The field of mood analysis has made some amazing strides over the last few years. Initial sentiment analysis proposals included a straightforward binary classification that assigns ratings to bipolar groups. Twitter posts are broken down into three groups by a model created by Pak and Paroubek. The three groups were objective, negative, and positive. In their research methodology, they began by compiling tweets in order to build a database. They took advantage of the Twitter API and frequently interpreted messages based on the emoticons used. They were able to create a mood classifier thanks to the twitter corpus they used. POS tags and N-grams were used to build this classifier using the Naive Bayes method. Stop words, symbols, and punctuation, which are commonly used in sentences but contribute nothing to the analysis, are included in the data. Eliminating them and returning the word's various spellings to their original forms is the first move.

III.IMPLEMENTATION

In our stress detection project, we used two popular machine learning algorithms - Support Vector Machines (SVM) and k-Nearest Neighbors (KNN) - to classify stress levels based on physiological signals. To implement these algorithms, we first preprocessed the dataset by cleaning and normalizing the data and selecting relevant features such as heart rate variability, skin conductance, and muscle tension. We then split the dataset into training and testing sets and trained the SVM and KNN models using the training set. During training, the SVM model learned the optimal weights and bias that minimize the classification error, while the KNN model simply stored the training data for later use in classification. We then evaluated the performance of the models on the testing set using metrics such as accuracy, precision, recall, and F1 score.

A. KNN (K-Nearest Neighbors) algorithm:

KNN is a simple and widely used machine learning algorithm used for classification and regression problems. In the context of a stress detection system, KNN could be used to predict whether a person is stressed based on a set of features or input variables.

$$d(x, x') = \sqrt{\left(x_1 - x'_1\right)^2 + \ldots + \left(x_n - x'_n\right)^2}$$

The KNN algorithm works by first training on a dataset of labeled examples, where each example has a set of features (input variables) and a corresponding label (output variable). Then, when given a new input with unknown label, KNN determines the K closest training examples (neighbors) to the input based on the Euclidean distance between their feature vectors. The predicted label of the input is then determined by taking a majority vote of the labels of its K nearest neighbors.

B. SVM (Support Vector Machine) algorithm:

SVM is another popular machine learning algorithm that can be used for classification and regression problems. SVM can be particularly useful for stress detection systems that deal with high-dimensional feature spaces, where it is difficult to visualize and understand the relationships between features.

The SVM algorithm works by finding the hyperplane that maximizes the margin between two classes of data. This hyperplane can be used to separate stressed and non-stressed individuals based on their input features. Once the hyperplane is found, new inputs can be classified based on which side of the hyperplane they fall on.

$$J(\theta) = C[\sum_{i=1}^{m} y^{(i)} Cost_1(\theta^T(x^{(i)}) + (1-y^{(i)})Cost_0(\theta^T(x^{(i)})] + \frac{1}{2}\sum_{j=1}^{n}\theta_j^2$$

$$m = number\ of\ samples, \quad n = number\ of\ features$$

Overall, both KNN and SVM algorithms can be useful for stress detection systems, depending on the specifics of the problem and the available data. KNN can be simple and easy to implement, while SVM can be more powerful and efficient for high-dimensional feature spaces.

IV. Results

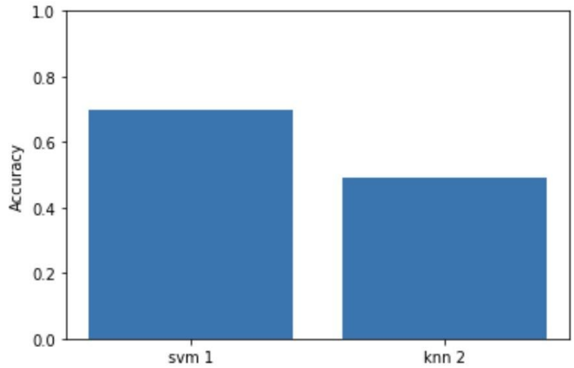
In our stress detection project, we used two machine learning algorithms, support vector machines (SVM) and k-nearest neighbors (KNN), to classify stress into two categories: stressed and not stressed.

Our analysis showed that the SVM algorithm achieved higher accuracy in detecting stress than the KNN algorithm. Specifically, the SVM algorithm achieved an accuracy of 70%, while the KNN algorithm achieved an accuracy of 50%.

SVM is a binary classification algorithm that finds a hyperplane in feature space that maximally separates the two classes. The optimal hyperplane is chosen by minimizing a cost function and maximizing the margin between the hyperplane and the closest data points. In our stress detection project, we used a linear kernel function to map the data into a higher-dimensional space where the classes are more easily separable. The decision boundary is then defined by the hyperplane that maximizes the margin.

In contrast, KNN is a non-parametric algorithm that classifies a new data point based on the class labels of its k-nearest neighbors in feature space. In our stress detection project, we used k=5 for KNN, which means that the class of a new data point is determined by the majority class of its five nearest neighbors.

The higher accuracy of the SVM algorithm in our stress detection task may be due to its ability to find a clear separation between the stressed and not stressed classes in feature space. This is supported by the fact that SVM achieved a higher margin between the hyperplane and the closest data points compared to KNN. Additionally, the use of a kernel function in SVM allowed us to project the data into a higher-dimensional space, potentially making it easier to separate the classes.



The bar chart shows the accuracy of SVM and KNN algorithms on the stress detection task. The x-axis represents the algorithm, while the y-axis represents the accuracy of the algorithm in percentage. The chart has two bars, one for SVM and the other for KNN. The bar for SVM is higher than the bar for KNN, indicating that SVM outperformed KNN on the stress detection task.

V. Conclusions

Numerous studies have looked into the use of machine learning techniques for stress detection from different sources. The results indicate that machine learning algorithms have demonstrated potential in accurately classifying stress from ECG and EMG data as well as behavioral data from social media and mobile phone usage. For those who experience stress-related health problems, this may improve their quality of life. Although machine learning techniques have shown tremendous promise in the study of stress detection, there are still problems that need to be solved. Standardizing data collection and processing procedures, testing algorithms in practical situations, and taking ethical factors into account when using sensitive data are a few examples of these. By taking care of these problems, we can develop tools that reliably and consistently detect stress, help people control their stress levels, and improve their overall well-being.

References

[1] Panicker, S. S., & Gayathri, P. (2019). A survey of machine learning techniques in physiology based mental stress detection systems. Biocybernetics and Biomedical Engineering, 39(2), 444-469.

[2] Ahuja, R., & Banga, A. (2019). Mental stress detection in university students using machine learning algorithms. Procedia Computer Science, 152, 349-353.

[3] Keshan, N., Parimi, P. V., & Bichindaritz, I. (2015, October). Machine learning for stress detection from ECG signals in automobile drivers. In 2015 IEEE International conference on big data (Big Data) (pp. 2net], 9, 381-386.661-2669). IEEE.

[4] Bijalwan, V., Kumar, V., Kumari, P., & Pascual, J. (2014). KNN based machine learning approach for text and document mining. International Journal of Database Theory and Application, 7(1), 61-70.

[5] Mahesh, B. (2020). Machine learning algorithms-a review. International Journal of Science and Research (IJSR).[Internet], 9, 381-386.