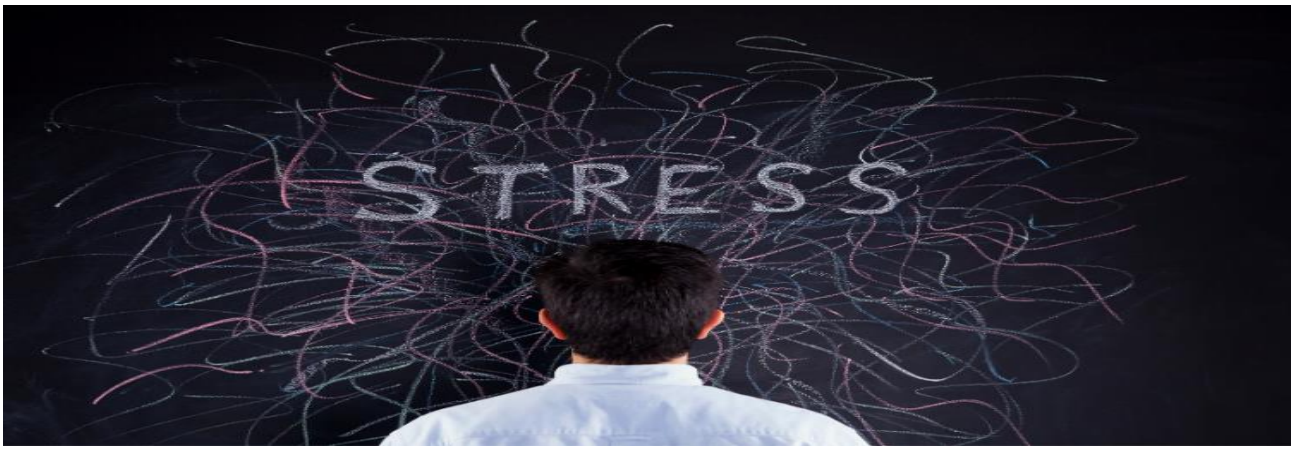
;



Stress Detection using Machine learning

Stress is a growing concern, as it has been linked to a variety of physical and mental health issues. In recent years, machine learning models have been developed to detect stress from various sources, such as physiological signals, facial expressions and speech. This literature survey discusses the current state of research in the field of stress detection using machine learning. The earliest work in this area focused on using physiological signals, such as electrocardiogram (ECG) or electroencephalogram (EEG) signals, to detect stress. These signals can be collected through wearable sensors or through contactless monitoring systems. For example, a study by Marcos et al. (2015) used a wearable sensor to monitor ECG signals and detected stress using a Support Vector Machine (SVM) model. Similarly, Gao et al. (2018) used a Convolutional Neural Network (CNN) to detect stress from EEG signals. Recent research has also explored the use of facial expressions for the detection of stress. Studies have used both static facial images and dynamic videos to detect stress. For instance, a study by Parveen et al. (2019) used a CNN to classify static facial images into stressed and non-stressed classes. Similarly, Li et al

Here we have used dataset from Kaggle. The dataset contains 2838 rows and 116 columns. Out of which we are only concerned about only two columns text and label. This dataset contains data posted on subreddits related to mental health. This dataset contains various mental health problems shared by people about their life. Fortunately, this dataset is labelled as 0 and 1, where 0 indicates no stress and 1 indicates stress.

Here we will use this dataset to detect stress from social media post such as Facebook, Instagram, twitter, LinkedIn etc.

Data preparation:

Since the data set contains lots of features and details, but in this case we are only concerned about the text and label columns. Here various steps are done to clean the text column. The stopwords, links, special symbols and language errors are removed.

After cleaning we must choose the most suitable model.

Model Selection:

Here the label column contains 0 and 1, which means it is a binary classification problem. To classify stress can use Naïve Bays, KNN, Decision tree etc. Here we will be using Bernoulli's Naïve Bays algorithm to classify stress and no stress.

Data Training and testing:

Here the data set is divided into 70:30 ratio. 70 percent of total data is used for training purpose and the remaining 30 percent is used for testing purposes.

Model accuracy: Model is able to make a quiet good prediction. The prediction score is 74.7. This could have been more better if we would have been a more data rows for more test cases.

Future Work:

We will try to make more accurate. For this we will consider a more robust dataset and also we will work with DL concepts to make it more useful.