Stress Level Detection Based on Sleeping Habits Using Decision Tree Classifier

Maria Jacob
PG Scholar
Amal Jyothi College of Engineering
Kottayam, Kerala
mariajacob20013@gmail.com

Binumon Joseph
Assistant Professor
Department of Computer Applications
Kottayam, Kerala
binumonjosephk@amaljyothi.ac.in

***Abstract*— In a fast-paced and demanding modern world, understanding and predicting human stress levels has emerged as a critical facet of promoting mental well-being. Considering today's lifestyle, people just sleep forgetting the benefits it provides to the human body. The reasons for not having a productive sleep could be many. Sleep disturbances are related to a number of physical, mental, and social problems. This study's main objective is to investigate how human stress might be detected using machine learning based on sleep-related behaviours. This study uses machine learning (ML) techniques to assess and forecast stress levels based on a person's sleeping patterns. To establish correlations with stress, this involves compiling a comprehensive dataset that includes sleep duration, quality, and other relevant factors.**

In addition to being a powerful tool for finding patterns and forecasts in large datasets, machine learning (ML) offers a promising new approach to stress assessment and management. This study includes the Decision Tree ML technique as a key component. It provides a clear and intelligible method for making decisions, which is helpful in comprehending the complex connections between stress and sleeping patterns. A structured model that is highly accurate in both classifying and predicting stress levels can be created using decision trees.

Keywords— Stress Level Detection, Sleeping Habits, Machine Learning, Decision Tree Classifier.

I. INTRODUCTION

Stress can be described as a condition of mental or emotional tension arising from challenging or inescapable situations. It can also be defined as a particular pressure exerted on the human body due to a variety of stress-inducing factors.

Stress is commonly classified as either negative or positive, with distress being classified as negative stress and eustress as positive stress. Acute stress, episodic acute stress, and chronic stress are the three different categories under which stress is categorized in terms of duration. Acute stress is often short-lived and transient, whereas episodic acute stress is characterized by recurrent acute stress episodes. The following categories apply to chronic stress: the deterioration of the stress reaction due to its activation repeatedly for a brief length of time, the human body becomes accustomed to stress and is unable to return to the ability to return to normal after extended exposure to stress, protracted stress, and abnormal stress response, i.e. a poor reaction. Long-term chronic stress exposure poses a number of health risks, including diabetes, obesity, sleeplessness, depression, and occasionally even cancer.

This seminar paper explores a potentially useful method for detecting stress levels based on sleep patterns. By utilizing the Decision Tree method and machine learning, we hope to develop a trustworthy and impartial way to evaluate a person's stress level by looking at their sleep patterns. This assignment is a good fit for the Decision Tree method, which is renowned for its interpretability and transparency. It helps us to visualize and comprehend the decision-making process that goes into predicting stress levels.

II. LITERATURE REVIEW

The idea of stress and how it affects people's health and wellbeing have been the subject of in-depth research over the years. Stress is a problem that affects people in all walks of life in today's society. Stress level prediction and understanding are becoming critical disciplines in data science, psychology, and medicine. We examine the corpus of research on stress prediction and the function of machine learning algorithms in this overview of the literature.

This paper [1] underscores the pervasive impact of stress on mental health, highlighting its recognition as a major concern by the World Health Organization. To mitigate the effects of stress, the paper introduces a system that employs statistical data and machine learning algorithms, including Decision Tree, Naïve-Bayes, and K-Nearest Neighbor, to accurately predict stress levels and enable proactive measures to address this significant issue.

This seminar paper [2] examines the vital function that stress plays in modern life and the importance of keeping an eye on stress levels for general health. Using the Random Forest Classifier method, it presents a revolutionary "Human Stress Detection Based on Sleeping Habits" system that can accurately predict stress levels by analyzing a variety of sleep-related factors. The method, which achieved a 100% training score and an impressive 97% test score, provides insightful information about stress management and has applications in medical research and personal health tracking.

Given the rising prevalence of stress-related issues among young people, this seminar paper [3] explores the importance of stress detection using machine learning techniques. It demonstrates the use of decision tree, random forest, and logistic regression techniques to examine a stress dataset obtained from Kaggle, with random forest obtaining the best accuracy. In light of real-time healthcare data, the conclusion recommends automating stress detection by sensor-based stress detection as well as additional advancements with k-fold cross-validation.

This seminar paper [4] combines a variety of machine learning methods, including Random Forest Classifier, Decision Tree, Naive Bayes, Support Vector Machine, and K-Nearest Neighbor, to detect stress based on human factors. The study seeks to identify potential health issues as well as the presence of stress by assessing a set of factors. The study highlights the potential for improving health monitoring and diagnosis by outlining future plans for stress categorization utilizing deep learning algorithms and real-time EEG signals. It also reports an amazing 93.2% accuracy rate for stress detection.

This seminar paper [5] addresses the critical issue of stress detection, utilizing machine learning techniques to enhance the quality of daily life. The study focuses on identifying stress levels in individuals during sleep using the SayoPillow dataset and offers various machine learning models. Notably, the SVM and Gaussian Naive Bayes models demonstrate superior performance, with F1-scores of 1.0, showing promise for early stress detection and potential applications in healthcare and IoT.

III. METHODOLOGY

The main aim of this system is to analyse and forecast stress levels using the DecisionTreeClassifier. Data collection, preprocessing, feature engineering, model selection, training, and evaluation are all included in the methodology.

A. Data collection

The foundation of our research lies in the quality and quantity of data collected. Our dataset, "Stress.csv," which includes pertinent physiological and sleep-related characteristics, was retrieved. Snoring Rate, Respiration Rate, Body Temperature, Limb Movement, Blood Oxygen, Eye Movement, Sleeping Hours, Heart Rate, Stress Level are the features of this dataset. The relevance of each item in the dataset was determined by its capacity to predict stress levels. The Pandas package was used to import the dataset, allowing for data manipulation and analysis.

B. Data Preprocessing

Even if raw data is large, it frequently has to be refined in order to reach its full potential. To minimize any inherent difficulties, such as missing values, outliers, and inconsistencies, we carefully handled the dataset throughout this phase. The process of preparing data included renaming columns to increase readability, conducting exploratory analysis of the data to determine the core properties of the data, and developing plans for dealing with missing data. To further reveal underlying patterns and correlations in the dataset, sophisticated data visualization techniques were used.

C. Machine Learning Model Selection

Choosing the right prediction model is a crucial step in our research process. A thorough evaluation of a number of machine learning algorithms was conducted, including the Gaussian Naive Bayes, Random Forest, DecisionTreeClassifier, Logistic Regression, Support Vector Machine, and k-Nearest Neighbors. The DecisionTreeClassifier was selected because it works well with both numerical and categorical data and can capture intricate feature interactions.

D. Training and Evaluation

After choosing a model, we separated the dataset into training and testing sets, with 80% of the data in the training set. In order for the DecisionTreeClassifier to learn from the data and identify patterns that connect characteristics to stress levels, it has to be carefully calibrated during the training phase. The foundation of our predictive model's capacity to extrapolate meaningful predictions and make broad generalizations from the data is this procedure.

E. Final Prediction

In the final stage of our methodology, we pivot from theory to practicality. We give an example of how our trained model may be used to forecast stress levels for fresh data, offering insightful and useful information. We provide a mapping of anticipated stress levels to human-readable labels to improve understanding and implementation, hence increasing the reach of the results.

This painstakingly crafted approach serves as the road map for our study, ensuring that each step is carefully carried out and openly reported. Our research aims to provide significant contributions to the field of stress level prediction and its practical applications by employing this all-encompassing methodology.

IV. BUILD MODEL

The model building is the main step in the stress level prediction. While building the model, user use the algorithms. The steps involved are:

A. Import Libraries

```
#Import necessary libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix
from sklearn.metrics import classification_report
from sklearn.tree import DecisionTreeClassifier
```

Fig.1 Import Libraries

B. Load Dataset

```
# Reading the CSV file 'SaYoPillow.csv' and storing the data in a DataFrame called 'data'
data = pd.read_csv("/content/Stress.csv")
```

Fig.2 Load Dataset

C. Data Exploration and Visualization

```
# Displaying the first 5 rows of the dataset
data.head()
# Displaying the last 5 rows of the dataset
data.tail()
```

Fig.3 Data Exploration and Visualization

D. Data Cleaning and Handling Missing Values

```
# Renaming the columns of the DataFrame for better readability and understanding
data.columns=['snoring_rate', 'respiration_rate', 'body_temperature', 'limb_movement', 'blood_oxygen',
          'eye_movement', 'sleeping_hours', 'heart_rate', 'stress_level']
data.head()
```

```
# To show statistical summary of the columns of our data
data.describe(include="all")
```

```
#checking for null values in the dataframe
data.isnull().sum()
```

Fig.4 Data Cleaning and Handling Missing Values

E. Data Splitting

```
# Split the data into features (X) and the target variable (y)
X = data.drop(['stress_level'], axis=1)
y = data['stress_level']
```

```
# Split the data into training and testing sets (80% training, 20% testing)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Display the shapes of the training and testing sets
print("X_train shape:", X_train.shape)
print("y_train shape:", y_train.shape)
print("X_test shape:", X_test.shape)
print("y_test shape:", y_test.shape)
```

Fig.5 Data Splitting

F. Model Training and Testing

```
decision_tree = DecisionTreeClassifier()
decision_tree.fit(X_train,y_train)
```

```
decision_tree.score(X_test,y_test)
```

```
y_predict = decision_tree.predict(X_test)

matrix = confusion_matrix(y_test, y_predict)

print("Confusion Matrix:")
print(matrix)
```

Fig.6 Model Training and Testing

G. Prediction

```
# Predicting Stress Levels
# To predict stress levels for new data, you can use the 'predict' method of the trained model.
# For example, let's assume we have new data in a DataFrame called 'new_data':
new_data = pd.DataFrame([[90.0, 23.0, 92.0, 15.0, 90.0, 95.0, 2.0, 70.0]], columns=X.columns)
```

```
predicted_stress_level = decision_tree.predict(new_data)
# Dictionary to map integer stress levels to human-readable labels
stress_level_labels = {
    0: "Low/Normal",
    1: "Medium Low",
    2: "Medium",
    3: "Medium High",
    4: "High"
}

# Assuming you already have the 'predicted_stress_level' from the previous code snippet
predicted_stress_label = stress_level_labels[predicted_stress_level[0]]

# Display the human-readable label for the predicted stress level
print("Predicted Stress Label for New Data:",predicted_stress_level[0],"(",predicted_stress_label,")")
```

Fig.7 Prediction

V. RESULT

A. Heatmap

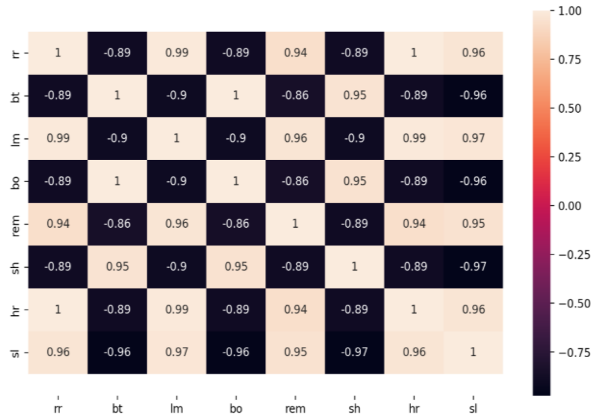

Fig.8 Heatmap

B. Evaluation of the Model

```
decision_tree = DecisionTreeClassifier()
decision_tree.fit(X_train,y_train)
```

```
▾ DecisionTreeClassifier
DecisionTreeClassifier()
```

```
decision_tree.score(X_test,y_test)
```

```
0.9841269841269841
```

```
report = classification_report(y_test, y_predict)

# Print the classification report
print("Classification Report:")
print(report)
```

```
Classification Report:
              precision    recall  f1-score   support

           0       0.96      1.00      0.98        23
           1       1.00      0.96      0.98        24
           2       1.00      1.00      1.00        28
           3       1.00      0.96      0.98        26
           4       0.96      1.00      0.98        25

    accuracy                           0.98       126
   macro avg       0.98      0.98      0.98       126
weighted avg       0.98      0.98      0.98       126
```

Fig.9 Evaluation of the Model

C. Final Result of the Model

```
# Predicting Stress Levels
# To predict stress levels for new data, you can use the 'predict' method of the trained model.
# For example, let's assume we have new data in a DataFrame called 'new_data':
new_data = pd.DataFrame([[90.0, 23.0, 92.0, 15.0, 90.0, 95.0, 2.0, 70.0]], columns=X.columns)

# Predict the stress level for the new data
predicted_stress_level = decision_tree.predict(new_data)
```

```python
# Dictionary to map integer stress levels to human-readable labels
stress_level_labels = {
    0: "Low/Normal",
    1: "Medium Low",
    2: "Medium",
    3: "Medium High",
    4: "High"
}
```

```python
# Assuming you already have the 'predicted_stress_level' from the previous code snippet
predicted_stress_label = stress_level_labels[predicted_stress_level[0]]

# Display the human-readable label for the predicted stress level
print("Predicted Stress Label for New Data:",predicted_stress_level[0],"(",predicted_stress_label,")")
```

Fig.10 Final Result of the Model

D. *Implementation*

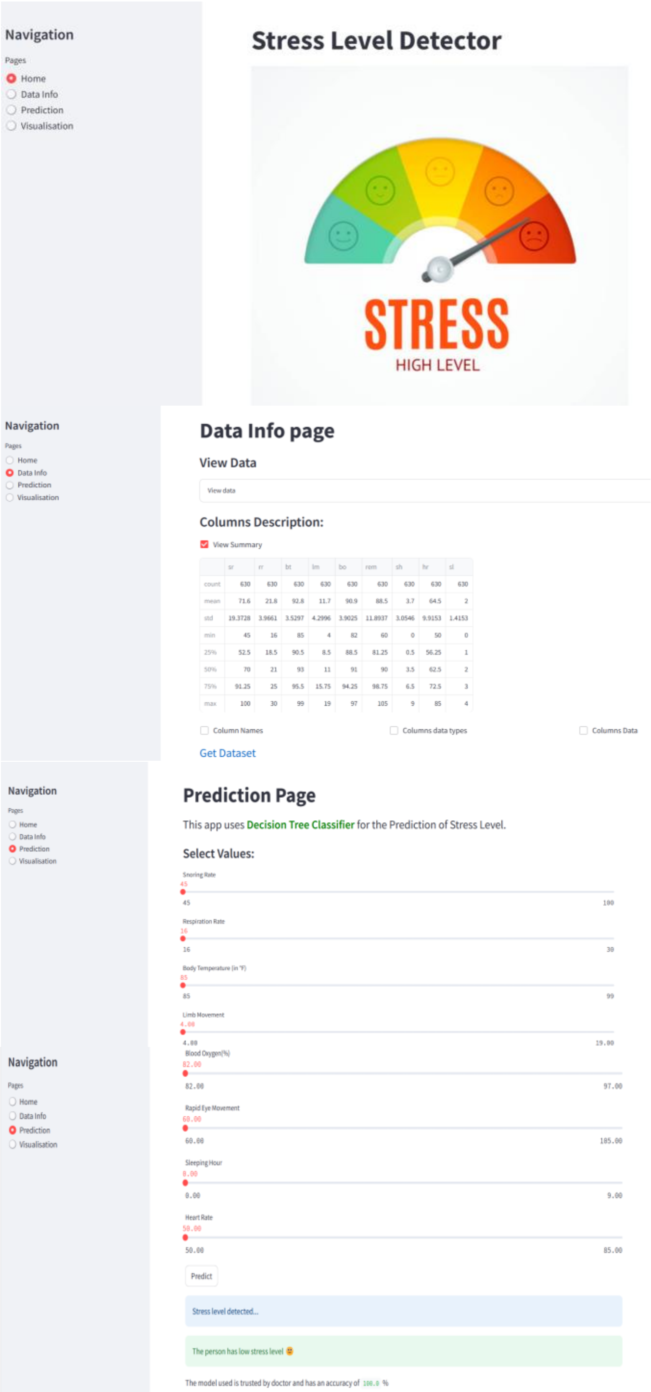

Fig.11 Implementation

VI. Conclusion

This seminar paper concludes by presenting a thorough methodology that includes all aspects of stress level prediction, from data pre-processing to model training and evaluation. With its interpretability and predictive ability, the Decision Tree classifier proves to be a useful tool in the field of stress treatment. The insights gathered from this study may help us identify and manage stress-related problems as we further explore the complex interactions between physiological variables and stress, which will eventually improve our quality of life and overall well-being.

VII. References

[1] V.R. Archana, "Stress Detection Using Machine Learning Algorithms", International Journal of Research in Engineering, Science and Management Volume-3, Issue-8, August-2020

[2] G. Jayawickrama, R.A.H.M. Rupasingha, "Human Stress Detection Based on Sleeping Habits Using Machine Learning Algorithms", 2023 3rd International Conference on Advanced Research in Computing (ICARC), IEEE Conference, 2023.

[3] R. Swarna Malika, "Stress Detection Using Machine Learning Techniques", Journal of Emerging Technologies and Innovative Research (JETIR)

[4] S.Kaviyab , S.Sarika c, "Stress Detection Based on Human Parameters Using Machine Learning Algorithms".

[5] Md Minhazur Rahman, ASM Mohaimenul Islam, Jonayet Miah, "sleepWell:Stress Level Prediction Through Sleep Data. Are You Stressed?"