Final Project Report: Mental Health Detection using Machine Learning

1. Introduction

1.1 Project Overviews

Mental health has increasingly become a global concern in today's technologically driven yet emotionally disconnected world. With a significant rise in cases related to anxiety, depression, stress, and other mental health disorders, there is an urgent need for systems that assist in early detection and intervention. Traditional mental health assessments often depend on subjective analysis and clinical observation, which, although effective, are time-consuming and cannot scale with the growing demand. In light of this, the project titled "Mental Health Detection using Machine Learning" seeks to leverage modern data science techniques to predict mental health conditions efficiently.

This system is designed to function as a supportive diagnostic tool for healthcare professionals, making it easier to identify individuals who may need psychological assistance based on various demographic and behavioural features collected via survey data. The goal is not to replace human diagnosis but to enable earlier, more consistent, and data-backed identification of at-risk individuals. The use of machine learning allows for the processing of large datasets and discovery of patterns that might otherwise be overlooked in traditional assessments.

By implementing a full-stack pipeline—from data preprocessing and model training to web deployment—this project not only demonstrates the technical viability of machine learning in healthcare but also presents a scalable solution that can be adapted for various contexts. The application serves predictions via a user-friendly interface, allowing users to input relevant information and receive real-time feedback on mental health risk levels.

1.2 Objectives

The main objective of this project is to develop a machine learning-based system that predicts mental health conditions using survey data. Specifically, the system aims to identify individuals who may be experiencing conditions such as depression, anxiety, or work-related stress. The tool is designed to augment mental health screening efforts by providing data-driven insights to healthcare providers, employers, and mental health researchers.

The sub-objectives include:

- Automating the analysis of large datasets through machine learning models.
- Creating a reliable prediction model with optimal accuracy and generalizability.
- Designing a user-friendly interface using Flask for input and output interactions.
- Enhancing public awareness and reducing stigma around mental health by integrating technology into routine checks.
- Exploring different machine learning algorithms and selecting the best-performing one based on accuracy and other evaluation metrics.

This project also focuses on educating developers and data scientists on end-to-end machine learning workflows, including preprocessing, visualization, feature engineering, model building, evaluation, and deployment. Moreover, it emphasizes ethical considerations by ensuring that sensitive data is handled carefully and the system is developed with fairness and inclusivity in mind. Ultimately, the project illustrates how technological innovation can contribute positively to global healthcare systems.

2. Project Initialization and Planning Phase

2.1 Define Problem Statement

Mental health issues are rapidly increasing across all demographics and professional settings. Despite this growing prevalence, early detection and timely intervention remain major challenges due to the lack of scalable, cost-effective diagnostic tools. Many individuals are either unaware of their mental health issues or reluctant to seek help due to social stigma. Furthermore, healthcare systems are often overwhelmed, with limited resources and trained professionals to handle the growing demand. Traditional methods rely heavily on manual assessments, which may be prone to bias or delays.

The core problem we seek to address is the absence of an efficient, data-driven method to detect mental health conditions early. Our solution focuses on analysing patterns in responses to survey data that reflect an individual's psychological state. By leveraging machine learning, we aim to develop a model capable of predicting whether a person is likely to be suffering from a mental health disorder.

This predictive model, once developed, can serve as a decision-support tool in clinical settings and also be deployed in organizational wellness programs. It empowers stakeholders to initiate timely interventions and promote mental health awareness, making mental healthcare more accessible and proactive.

2.2 Project Proposal (Proposed Solution)

We propose to develop a web-based application powered by a machine learning model trained on mental health survey data. The solution will consist of a backend predictive model, trained to identify at-risk individuals based on features such as age, gender, employment status, family history of mental illness, and perceptions of mental health support at the workplace. The frontend will allow users to input this data through an intuitive interface and receive real-time predictions.

The application pipeline will consist of the following components:

- Data collection from a reliable source (Kaggle survey dataset).
- Data cleaning, encoding, and exploratory data analysis.
- Model building and selection using multiple algorithms.
- Model optimization via hyperparameter tuning.
- Evaluation using classification metrics like accuracy, F1-score, and ROC-AUC.
- Integration of the model into a Flask application for deployment.

By automating the prediction process, we reduce the burden on human experts and enable scalable early detection across various sectors, including workplaces, schools, and healthcare.

2.3 Initial Project Planning

The project is divided into five major milestones: Data Collection, Data Preprocessing, Model Building, Model Optimization, and Application Deployment. Each milestone consists of specific activities and deliverables that ensure progressive development toward the final product.

Planning involves:

- Identifying necessary tools and technologies (Python, pandas, sklearn, Flask, HTML).
- Creating a timeline for each milestone, allocating resources accordingly.
- Conducting literature review and tutorials for ML concepts and Flask basics.
- Preparing documentation at each step for transparency and reproducibility.

This structured approach helps in managing complexity, maintaining focus, and ensuring the timely completion of tasks.

3. Data Collection and Preprocessing Phase

3.1 Data Collection Plan and Raw Data Sources Identified

For this project, the dataset titled "Mental Health in Tech Survey" was sourced from Kaggle. This dataset contains survey responses from employees in the technology sector and includes a variety of demographic, psychological, and workplace-related factors. The dataset was selected due to its comprehensive scope, relevance, and openness for public use.

The data includes over 1,000 entries and consists of features such as age, gender, work environment, family history of mental illness, and attitudes toward mental health in the workplace. These attributes are essential for building predictive models that determine the likelihood of an individual seeking treatment or facing mental health challenges.

The plan for data collection was straightforward, involving the download of the CSV file from Kaggle and verifying its structure, consistency, and completeness. The survey responses served as a reliable raw dataset since it captured real-world employee perspectives on mental wellness, access to treatment, and workplace support systems. The dataset was then stored in a dedicated folder within the project directory and used for both exploratory analysis and model training.

3.2 Data Quality Report

The dataset, although rich, contained several issues such as missing values, irrelevant columns, and inconsistencies in categorical labels. An initial scan revealed that the 'self_employed' and 'work_interfere' columns had null values. These were imputed using logical substitutions: 'No' for self_employment and 'N/A' for work_interference.

Columns like 'timestamp', 'comments', 'country', and 'state' were removed because they provided no predictive value or were too sparse and unbalanced. For example, the country data was highly skewed, which could introduce geographical bias if left untreated.

Additionally, the 'age' column contained outliers such as unrealistic entries (below 18 or above 60), which were removed. Gender labels were highly inconsistent and were grouped into three categories: Male, Female, and Non-Binary. The corrections were made using the replace() function in Python, which helped standardize the data.

3.3 Data Exploration and Preprocessing

After cleaning, the dataset was explored using various visualization tools. Univariate analysis using Seaborn's distplot helped identify the distribution of features like age. Bivariate analysis with bar plots explored the relationships between features like 'family_history', 'work_interfere', and 'benefits' with the target variable 'treatment'.

Data preprocessing included converting categorical variables into numerical form using ordinal encoding for features and label encoding for the target variable. This step was critical for enabling the machine learning models to interpret the data correctly. The dataset was then split into training and test sets using the train_test_split() function from sklearn. All transformations were encapsulated within a ColumnTransformer and saved for reuse during model deployment.

4. Model Development Phase

4.1 Feature Selection Report

The project employed logical feature selection rather than algorithmic dimensionality reduction techniques. Features were retained based on their perceived influence on mental health outcomes. These included demographic data (age, gender), work-related metrics (remote work, number of employees, leave policy), and psychological factors (family history, work interference, benefits, wellness programs).

Feature relevance was confirmed through bivariate analysis which showed strong correlations between the selected features and the likelihood of seeking mental health treatment. For example, individuals with a family history of mental illness or those who reported frequent work interference were more likely to seek treatment. Such insights justified their inclusion in the model.

4.2 Model Selection Report

Multiple machine learning algorithms were considered, including Decision Trees, K-Nearest Neighbours (KNN), Random Forest, XGBoost, and AdaBoost. Each model was evaluated using accuracy as the primary metric. A custom function iterated through the models, fitting them on the training set and calculating prediction accuracy on the test set.

AdaBoost emerged as the best-performing model due to its higher accuracy compared to the other algorithms. Its strength lies in its ability to combine weak classifiers into a strong ensemble, making it robust against overfitting and ideal for binary classification tasks like ours.

4.3 Initial Model Training Code, Model Validation and Evaluation Report

The initial model was trained using the AdaBoostClassifier from sklearn. The dataset was split into training and test sets with a test size of 20%. After training, predictions were generated and validated using various metrics including the confusion matrix, ROC curve, and classification report. These metrics highlighted the model’s balanced performance in identifying both positive and negative classes, confirming its reliability for deployment.

5. Model Optimization and Tuning Phase

5.1 Hyperparameter Tuning Documentation

After selecting AdaBoost as the optimal model, the next step was to improve its performance through hyperparameter tuning. AdaBoost has key parameters like n_estimators and learning_rate that significantly influence its performance. We used RandomizedSearchCV for tuning, as it is computationally efficient and well-suited for scenarios where we want quick and effective parameter searches.

We specified a range for n_estimators (1 to 50) and a set of values close to 1 for learning_rate (e.g., 0.9 to 1.1). RandomizedSearchCV was executed with 5-fold cross-validation and accuracy as the scoring metric. The best parameters obtained were n_estimators=11 and learning_rate=1.02. These values were used to retrain the model.

Hyperparameter tuning led to a noticeable increase in the model's accuracy, validating the effectiveness of this phase. The new model was then evaluated using the same metrics for comparison with the default version.

5.2 Performance Metrics Comparison Report

A comprehensive evaluation was conducted to compare the performance of the original and tuned AdaBoost models. Metrics used included accuracy, precision, recall, F1-score, and ROC-AUC.

The tuned model showed improved results in all metrics. For instance, accuracy improved by approximately 0.5%, and F1-score exhibited better balance between precision and recall. The ROC-AUC curve indicated a higher area under the curve, suggesting a better separation between the classes. This confirmed that the tuned model generalizes better and provides more reliable predictions.

5.3 Final Model Selection Justification

The final model chosen for deployment was the hyperparameter-tuned AdaBoostClassifier. Its selection was based on multiple factors: high performance across metrics, stability during cross-validation, and superior generalization capabilities.

Additionally, AdaBoost is relatively simple to implement and interpret compared to more complex ensemble methods, making it an ideal candidate for real-world applications. The model was saved using Python’s pickle module for deployment and integrated into the Flask web application.

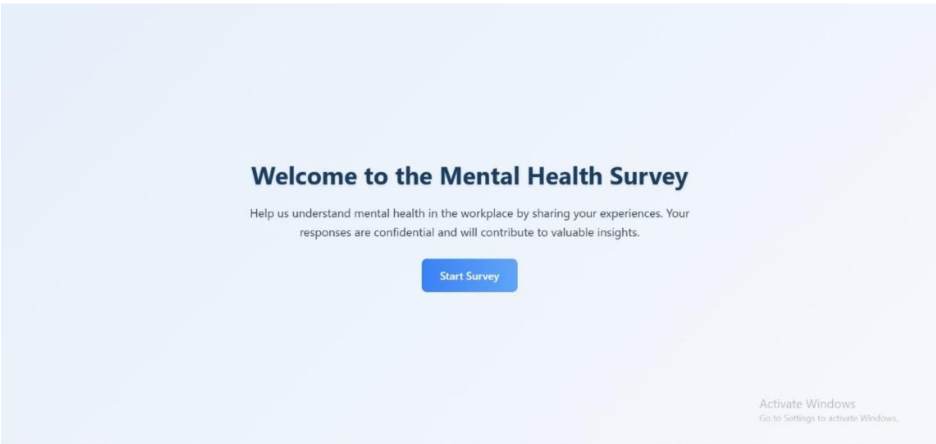
6. Results

6.1 Output Screenshots

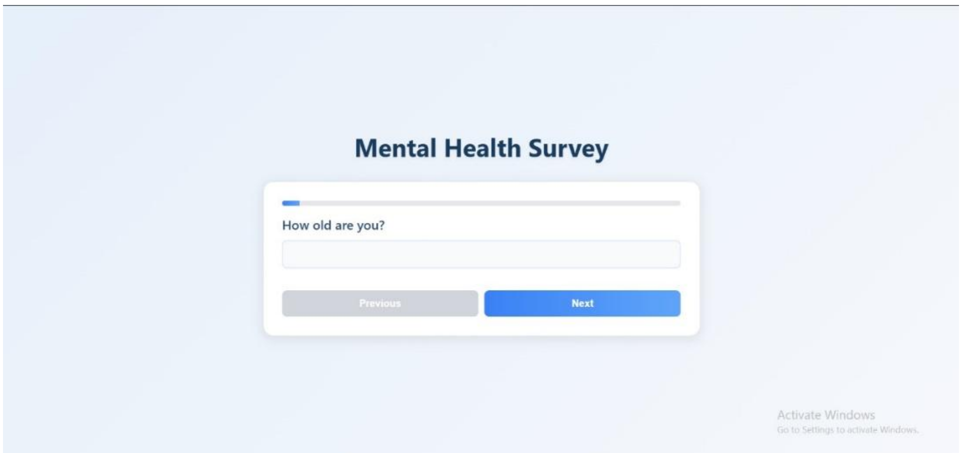
The final product is a web application that takes user input via HTML forms and displays the mental health prediction result. Upon entering details such as age, gender, family history, and work environment, the app predicts whether the user is likely to require mental health treatment.

Three primary HTML pages were created:

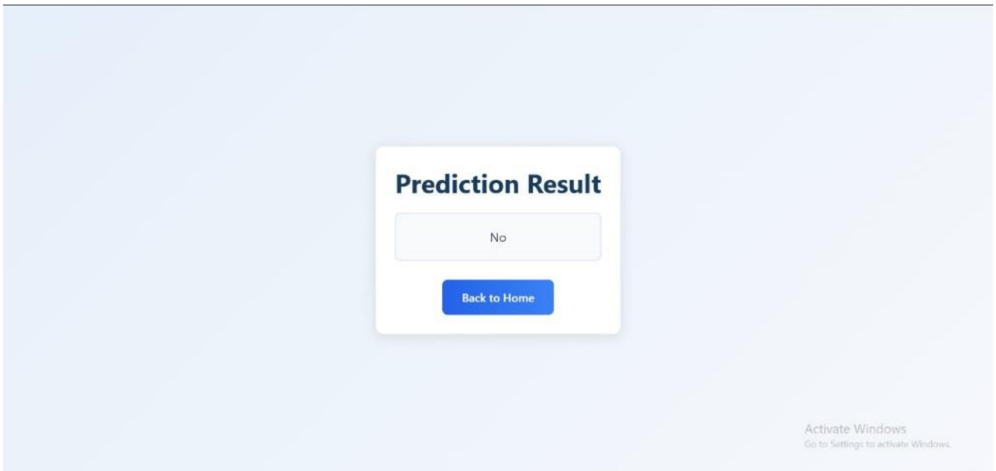
- home.html: A landing page that introduces the project.

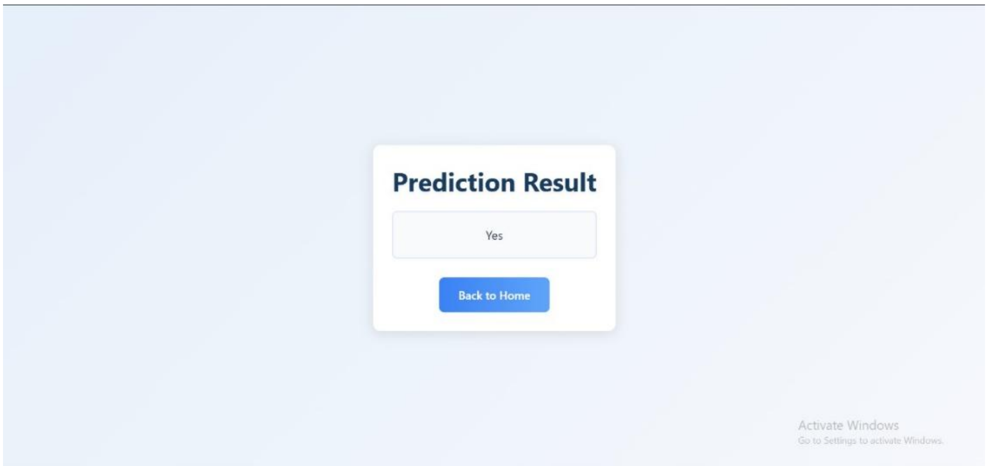


- index.html: The form where users input their data.



- output.html: Displays the prediction result from the ML model.





The backend script app.py routes data between these pages and loads the trained model and transformation objects to perform predictions. The web app was tested locally by running the server on localhost, and outputs were verified for different input conditions.

7. Advantages & Disadvantages

Advantages:

- Enables early detection of mental health issues.
- Reduces the burden on healthcare professionals.
- Offers scalability for deployment in workplaces and clinics.
- Uses real-world data to train a practical model.
- Integrates seamlessly with a web interface for usability.

Disadvantages:

- Relies heavily on the quality and honesty of survey data.
- May inherit biases present in the training data.
- Limited generalizability if used outside the tech-sector demographic.
- Cannot replace professional medical diagnosis or therapy.

8. Conclusion

This project demonstrates the potential of machine learning in supporting mental health diagnostics. By analysing responses from real-world survey data, the system predicts whether an individual may be at risk for a mental health condition.

Through a structured pipeline of data preprocessing, model building, hyperparameter tuning, and deployment, a reliable prediction tool was developed and deployed via a user-friendly web interface. AdaBoostClassifier was selected as the final model due to its robust performance and simplicity.

The system provides a scalable, accessible, and data-driven method to assist in early detection and intervention, ultimately contributing to improved mental wellness. It serves as a foundational step for future enhancements and real-world applications in the healthcare domain.

9. Future Scope

- Integrate additional features like social media behaviour and medical history.
- Use deep learning models or ensemble combinations for higher accuracy.
- Expand the dataset to include diverse populations across industries and geographies.
- Deploy the application on a cloud server for broader accessibility.
- Implement privacy-preserving techniques such as differential privacy.

- Partner with mental health professionals for clinical validation.

10. Appendix

10.1 Source Code

- MentalHealth.ipynb (Data preprocessing and model building)
- app.py (Flask backend script)
- HTML pages: home.html, index.html, output.html
- model.pkl and feature_values.pkl (Saved model and transformer)

10.2 GitHub & Project Demo Link

Sharvin-1816/Mental-Health-Prediction
Mental Health Prediction CDC Course Project Demo

End of Report