AI Model Report: Design and Deployment of an AI-Powered Predictive System

1. Introduction

This report details the "Design and Deployment of an AI-Powered Predictive System," a capstone project developed as part of the AI/ML Class under the Digital Skillup Africa program, coordinated by The Incubator Hub. This project demonstrates proficiency in developing and deploying an artificial intelligence solution from inception to a functional application.

- **Author**: Joshua Oluwatobiloba Iseyemi
- **Program**: Digital Skillup Africa
- **Coordinators**: The Incubator Hub
- **Instructors**: Mr. Oluwole Olajide, Mr. Blessing Olorunfemi, Mr. Isaac Oyekunle

2. Project Overview

The project leverages data from the UCI Machine Learning Repository, focusing on student performance in Portuguese schools. The primary objective of this AI model is to predict student performance in mathematics based on various features. The project specifically utilizes the mathematics dataset for model development and evaluation.

3. Data Description

The dataset used for this project is student-mat.csv, which contains 33 columns and 395 entries. It includes diverse student attributes such as:

- Demographics (e.g., age, sex, address, family size)
- Parental information (e.g., education, jobs, status)
- School-related factors (e.g., travel time, study time, failures, school support)
- Social aspects (e.g., family relationship quality, free time, going out, alcohol consumption, health)
- Academic grades (G1 - first period grade, G2 - second period grade, G3 - final grade)

The dataset was loaded and inspected for initial statistical analysis and data integrity.

4. Methodology

4.1. Data Preprocessing and Feature Engineering

The initial steps involved loading the dataset and performing a statistical analysis to understand the data distribution and identify potential areas for preprocessing. Categorical features within the dataset were converted into numerical representations using one-hot encoding to make them suitable for machine learning algorithms. Furthermore, feature scaling was applied to normalize numerical features, which is crucial for optimizing model performance.

4.2. Model Selection and Training

The project utilizes a Random Forest Regressor model for predicting student final grades (G3). The dataset was split into training and testing

sets, with 80% allocated for training and 20% for testing, ensuring that the model's performance could be evaluated on unseen data.

4.3. Model Evaluation

The model's performance was rigorously evaluated using key regression metrics:

- **Mean Absolute Error (MAE):** This metric quantifies the average magnitude of the errors in a set of predictions, without considering their direction.
- **Mean Squared Error (MSE):** MSE measures the average of the squares of the errors, giving higher weight to larger errors.
- **Root Mean Squared Error (RMSE):** RMSE is the square root of MSE and provides an error metric in the same units as the target variable.
- **R-squared (R2) Score:** R2 indicates the proportion of the variance in the dependent variable that is predictable from the independent variables.

After evaluation, the trained model and the scaler were saved as a `math_model.pkl` file using `joblib` for future deployment.

5. Model Deployment

5.1. Backend (app.py)

The model is deployed as a web service using Flask, a Python web framework. The `app.py` script serves as the backend, handling requests for predictions. It performs the following functions:

- Loads the pre-trained `math_model.pkl`, which includes both the scaler and the predictive model.
- Exposes a `/predict` endpoint that accepts POST requests containing student features in JSON format.

-
-
  Validates incoming data, ensuring exactly 9 features are provided and are numerical.
-
-
  Splits categorical and numerical features for appropriate processing.
-
-
  Applies the loaded scaler to transform the numerical features.
-
-
  Uses the model to generate a prediction for the final grade (G3).
-
-
  Returns the prediction as a JSON response.
-
-
  Includes a root endpoint (/) to serve the frontend index.html file.
-
  5.2. Frontend (index.html)
  The user interface is a web page (index.html) that allows users to input student data and receive predictions. Key features of the frontend include:
-
  A user-friendly form with input fields for relevant student features such as age, parental education, travel time, study time, past failures, social outings, and previous grades (G1, G2).
-
-
  Client-side validation to ensure all fields are filled with valid numerical inputs.
-
-
  An asynchronous JavaScript function to send input data to the backend /predict endpoint.
-

- A dynamic display of the predicted final grade (G3) to the user.
-
- Visual feedback for loading states and error messages.
-

6. Conclusion

This project successfully demonstrates the end-to-end process of developing an AI model for predicting student performance and deploying it as a web application. The robust methodology, from data preprocessing to deployment with a user-friendly interface, highlights a comprehensive understanding of AI/ML principles and their practical application.

7. Contact Information

For further inquiries, please use the following:

- **Email**: iseyemijoshua002@gmail.com
-
- **Kaggle**: joshuaiseyemi
-
- **GitHub**: espejo002