Project Proposal (Artificial Intelligence)

Title: Student Admission Predictor with Smart Recommendation System

Submitted by:

Name: Md. Sami Alam

ID: 222-16-656

Section: 15_16_A

Subject Code: CIS 315

Course Title: Artificial Intelligence

Semester: Spring 2025

Department Of **CIS**, **DIU**

Due Date: 9th April, 2025

Group Members ID's: 222-16-657, 222-16-672, 222-16-681, 222-16-673

Submitted to:

Name: Md. Mizanur Rahoman

Department: Computing and Information System, DIU

Submission date 7th April , 2025

Acknowledgment

I would like to express my heartfelt gratitude to Allah for providing me with the strength and guidance throughout the completion of this project.

I would like to extend my special thanks to Md. Mizanur Rahoman, our honorable teacher, for giving me the invaluable opportunity to work on this exciting and challenging semester project. His constant guidance and encouragement helped me develop a deep understanding of the AI-based Student Admission Predictor, allowing me to explore various aspects of machine learning, data visualization, and recommendation systems. This project has enriched my knowledge and skills in both the theoretical and practical application of these technologies.

I would also like to thank my parents for their unwavering support, encouragement, and understanding throughout this project. Their constant belief in me helped me stay focused and complete this project within the given time frame.

Lastly, I would like to express my gratitude to my classmates who have been a source of constant help and motivation. Their insights, advice, and collaboration were instrumental in overcoming various challenges during the project.

Background and Motivation

In the highly competitive landscape of higher education admissions, students often face uncertainty about their chances of being accepted into their desired universities. Many are unaware of which academic or personal components (such as GRE, TOEFL, CGPA, and research experience) most influence their chances. This project aims to solve that problem using Machine Learning to predict admission probability and provide personalized, actionable recommendations.

Project Objectives

The primary objective of the **Student Admission Predictor** project is to predict a student’s probability of admission to a university based on key input features: GRE score, TOEFL score, CGPA, and Research experience. The model uses **logistic regression**, a powerful algorithm for binary classification, to provide accurate predictions of admission chances, helping students understand their likelihood of acceptance into graduate programs.

An important aspect of this project is the visualization of feature contributions, which provides a detailed breakdown of how each input feature influences the admission prediction. Through bar charts or graphs, students can see which areas of their academic profile are strong and which need improvement, enabling them to make data-driven decisions.

The project also aims to offer smart recommendations, guiding students on how to improve their admission chances. For example, the system will suggest increasing GRE scores or improving research experience to boost the overall admission probability.

A user-friendly web interface is essential to make the tool accessible to students. Using modern web technologies like React and Flask, the platform ensures seamless interaction with the model and provides an intuitive experience for users.

Lastly, the project is designed to be scalable, modular, and easily integrable with other academic advising tools, allowing universities to adopt and adapt it to their needs in the future.

Features of the Student Admission Predictor

The **Student Admission Predictor** project offers several key features designed to provide valuable insights for students seeking to predict their likelihood of admission into graduate programs. Below are the core features:

1. Admission Prediction

The primary feature of the project is the Admission Prediction tool. It calculates the probability of a student's admission to a university based on four critical input features: GRE score**,** TOEFL score**,** CGPA, and Research experience. The model uses a logistic regression algorithm, a binary classification technique, to predict whether a student will be admitted or not. The output is a probability score that reflects the likelihood of admission, helping students understand their chances in a data-driven manner.

2. Feature Contribution Visualization

The Feature Contribution Visualization feature allows students to see how each of the four academic factors—GRE score, TOEFL score, CGPA, and Research experience contributes to the final admission prediction. A bar chart is generated, displaying the impact of each feature on the admission probability. This feature is vital because it helps students identify their strengths and weaknesses, showing them which areas they need to improve to increase their chances of admission. For instance, if the GRE score has a high contribution to the prediction, improving that score could enhance their admission likelihood.

3. Smart Recommendations

To make the tool even more insightful, the project also includes Smart Recommendations. Based on the prediction and feature contribution analysis, the system offers personalized suggestions on how to improve a student’s chances of admission. For example, if the model suggests that the GRE score is a major factor in the decision, the system may recommend preparing for a GRE retake to achieve a higher score, thereby improving the student's overall admission probability.

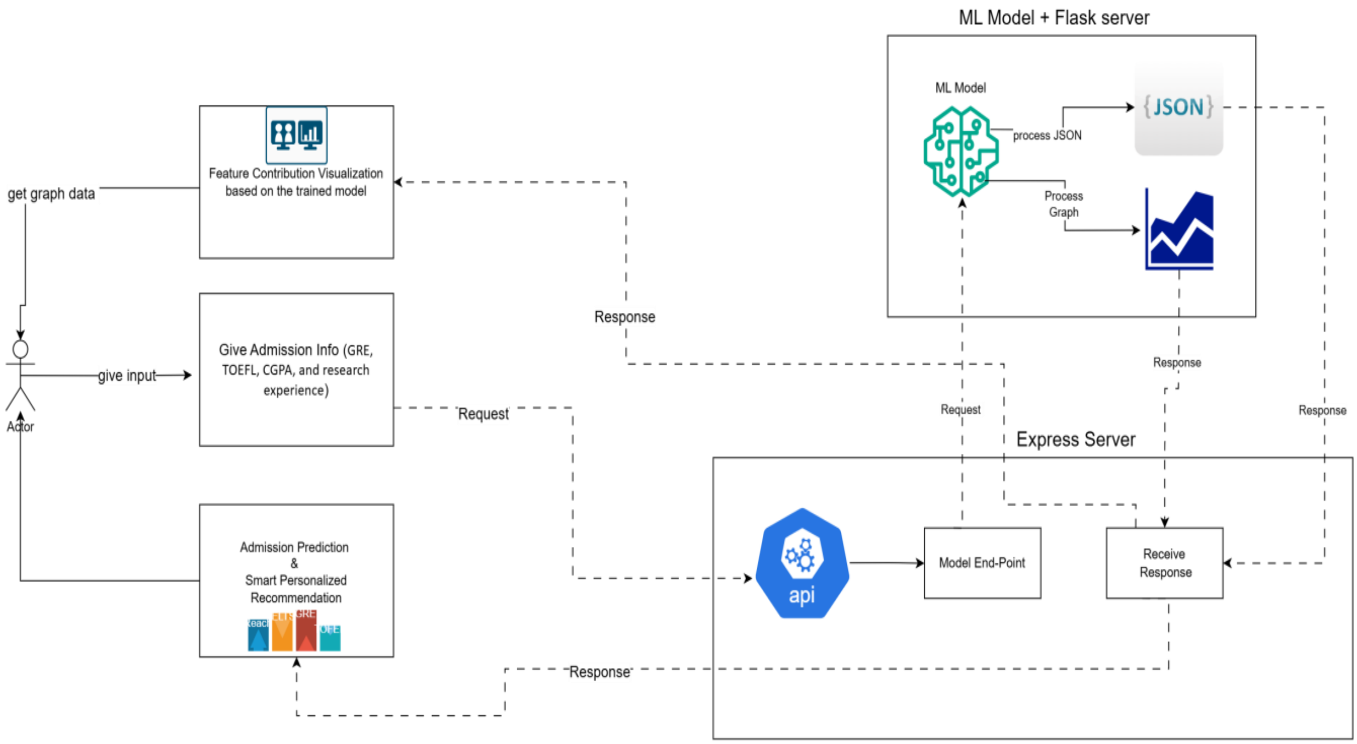
Model Development

The core of this project is a **Logistic Regression** model that predicts a student's likelihood of admission based on four key features: GRE Score (out of 340), TOEFL Score (out of 120), CGPA (out of 4.0), and Research Experience (binary, 0 = No, 1 = Yes). The model computes the probability of admission and uses the coefficients of each feature to determine its impact on the prediction. These contributions are visualized in a bar chart, helping users understand which factors influence their chances the most.

Model Training Process:

1. **Preprocessing the Dataset:** The dataset, stored in a dataset.csv file, is first cleaned and normalized. Missing data is handled, and numerical values are scaled to ensure uniformity and improve model performance.
2. **Training the Model:** Using Scikit-learn, the logistic regression model is trained on the processed dataset. The model learns the relationships between the features and admission outcomes.
3. **Model Evaluation:** The model is evaluated using accuracy and precision metrics to assess its prediction capability.
4. **Saving the Model:** After training, the model is saved as a .pkl file for easy future use.

System Architecture



For Better Visualization Click this Link:

https://drive.google.com/file/d/1HPLzND0vOmBLKEo7-JSjfgEquqtLPUYR/view?usp=sharing

Technology Stack

Frontend:

React.js: JavaScript library for building the user interface.

Axios: For making HTTP requests from react to the express backend.

Backend:

ExpressJs: NodeJs framework for CORS and For talking to react and forwarding data

Flask: Python web framework for creating the RESTful API.

Scikit-learn: For building and using the Logistic Regression model.

Machine Learning Libraries:

NumPy: For numerical operations and data handling.

Matplotlib: For generating and saving the feature contribution graphs.

Thank You