Department of Computer Science,

Northeastern Illinois University,

Chicago, Illinois

Master's Project Report

Developing Machine Learning-Enhanced AI Models for Underrepresented Transfer Students Counseling System (ACOSUS)

By

Sanjana Motte Vishwanatha

Under the guidance of

Dr. Xiwei Wang, Department Chair & Project Advisor

Dr. Ken Sotak, Academic Advisor & Co-Advisor

Dr. Lizi Zhu, Third Faculty Member

December 2024

Acknowledgments:

The development and progress of the ACOSUS research project would not have been possible without the invaluable support and guidance of many individuals and organizations.

I extend my heartfelt gratitude to my mentor, Dr. Xiwei Wang, whose mentorship extended beyond the academic realm, providing comfort, and motivation during tough times, and for his unwavering support throughout my research journey. His deep commitment to academic excellence and meticulous attention to detail have significantly shaped this research. I am equally thankful to my second and third advisors, Dr. Ken Sotak and Dr. Lizi Zhu, for their constructive feedback and essential suggestions that enhanced the quality of my work.

To my research partners, Deep Mandloi and Nikunj Patel, your endless patience and understanding, especially during the most demanding phases of this research, have been my anchor.

I must also express my deepest appreciation for my family, who stepped in not only with emotional reassurance but also with their unconditional love and sacrifice, which have been the foundation of my resilience and success.

Finally, my thanks go out to the NSF(National Science Foundation), institutions, and data providers who have facilitated access to critical data resources, allowing me to craft a robust and meaningful model.

Table of Contents:

Table of Figures 5

Abstract: 6

1. Introduction 7

2. Literature Review 10

2.1. Overview of Challenges Faced by Underrepresented Transfer Students: 10

2.2. Integration of AI and Technological Solutions in Higher Education Counselling: 10

2.3. Innovative Data Analysis Methods for Understanding Transfer Student Concerns: 11

2.4. Combining Survey-Based and NLP Approaches to Understand Transfer Student Needs: 11

2.5. Relevance to ACOSUS System Design: 11

3. Methodology: 13

3.2. Input Data Structure and Parameters: 14

3.3. Data Seeding: 16

3.4. Numerically Converted Input Parameters for ACOSUS Model: 17

3.5. Data Cleaning: 17

3.6. Normalization of the processed input data: 18

4. Proposed System: 21

4.1. System Architecture: 21

4.2. Model Architecture: 23

4.2.1. Defining the Readiness Score: 23

4.2.2. Train - test split: 23

4.2.3. Model Evaluation Metrics: 24
4.3. Regression Model: 25
4.3.1. Introduction: 25
4.3.2. Performance Analysis: 26
4.4. Neural Network: 29
4.4.1. Introduction: 29
4.4.2. Training - Performance Analysis: 31
4.4.3. Prediction – Performance Analysis: 33
4.5. Connecting to OpenAI API: 35
5. Conclusion 37
6. Future Work 38
7. References: 40

Table of Figures

Figure 1: Process of data collection, cleaning, visualizing, and identifying topics 12
Figure 2: Workflow Diagram 13
Figure 3: System Architecture 23
Figure 4: MSE, MAE, R2 vs Alpha for Regression 27
Figure 5: Learning curve for Regression 28
Figure 6: Defining the Neural Network 31
Figure 7: MSE, MAE, R2 vs Epochs for the Best Hyperparameter values 32
Figure 8: MSE, MAE, R2 vs Epochs for a random Hyperparameter value 33
Figure 9: Residual Plot 35
Figure 10: GPT Prompt 36
Figure 11: Sample Output 36

Abstract:

This paper introduces the ACOSUS platform (AI-Driven Counselling System for Underrepresented Transfer Students), which aims to support underrepresented students transitioning into STEM (Science, Technology, Engineering, and Mathematics) majors. ACOSUS offers personalized guidance, predictive analytics, and real-time assistance to help students navigate the academic and career challenges they face when moving from community colleges to universities.

The project is aimed at developing a robust predictive model to calculate the success rates of students based on a range of educational and personal metrics. By leveraging the power of machine learning, specifically neural networks and regression models, which aim to provide accurate predictions that can be useful for STEM students to tailor their interventions and resources effectively.

ACOSUS is built by data collection, utilizing a comprehensive approach, gathering data through a web API. This data included variables such as GPA, SAT scores, course loads, family and demographic information, and other personal attributes that influence academic outcomes. TensorFlow and Keras were employed to construct and train the models. The Flask framework was instrumental in deploying this model as a web application, allowing for real-time data input and prediction retrieval.

A pivotal enhancement to the project was the integration of OpenAI technology to gather detailed feedback on student performance. This integration allows the predictive model to not only assess success rates but also provide personalized feedback and recommendations. With continued research, development, and implementation, ACOSUS strives to significantly improve student retention and success in STEM fields.

Keywords: ACOSUS, Underrepresented Students, STEM Majors, Counselling System, Predictive Analytics, Regression, Neural Networks, Student Support.

1. Introduction

In higher education, the transition from community colleges to universities represents a critical yet often challenging step for many underrepresented students pursuing degrees in Science, Technology, Engineering, and Mathematics (STEM). These students encounter a wide array of obstacles, including academic readiness, financial limitations, social integration issues, and uncertainties surrounding future career paths. To address these unique challenges and provide tailored guidance, the AI-Driven Counseling System for Underrepresented Transfer Students (ACOSUS) offers a groundbreaking approach aimed at supporting these students through this crucial phase of their academic journey.

This system leverages a sophisticated AI infrastructure comprising predictive analytics, regression analysis, and neural network models to offer personalized guidance and enhance students' academic and career trajectories.

At the heart of ACOSUS is a suite of advanced machine-learning models that analyze vast arrays of student data to assess academic readiness and predict future outcomes. These models utilize a comprehensive range of inputs, including academic performance, demographic factors, and personal engagement metrics, ensuring a deep understanding of the unique challenges each student faces.

Predictive analytics within ACOSUS proactively identify potential academic and social obstacles, allowing for timely interventions tailored to individual student needs. Neural network models in ACOSUS are designed to detect complex patterns within the data, enabling predictions on key outcomes such as potential career success. These models are continually updated with new data, which enhances their accuracy and the system’s ability to adapt to evolving educational dynamics.

Furthermore, ACOSUS is designed to complement, not replace, the role of human instructors in providing personalized support to students. By offering instructors detailed insights into students' strengths, weaknesses, and aspirations, the system enables them to deliver more focused and effective guidance tailored to individual needs. In addition to enhancing instructor support, ACOSUS empowers students by giving them convenient access to instructional resources and support services, promoting a culture of self-directed learning and fostering greater independence. This dual approach strengthens both personalized learning and student empowerment, helping students navigate their academic journeys more effectively.

This report provides a comprehensive overview of the ACOSUS system, covering its core functionalities, backend implementation, data collections, model evaluation, and OpenAI implementations. Additionally, we explore potential future enhancements to expand the effectiveness and reach of ACOSUS. Through ongoing collaboration and innovation, ACOSUS seeks to transform student support services, foster greater diversity and inclusion in STEM education, and ultimately empower students to succeed both academically and professionally.

The thesis is systematically organized into several chapters, each providing detailed insights into various aspects of the ACOSUS system.

- Chapter 1 sets the stage for the report, explaining the purpose and significance of the ACOSUS system. It provides a background on the support needed by underrepresented STEM transfer students and the role of the ACOSUS system in addressing these needs.

- Chapter 2 talks about the existing literature related to the challenges faced by underrepresented transfer students and is examined.

- Chapter 3 tells detailed methodologies used in the project outlined here, including the structure of input data, parameters, data seeding, numerically

converted input parameters for the ACOSUS model, data cleaning, and normalization of the processed input data.

- Chapter 4 describes the system and model architecture of ACOSUS, detailing the readiness score, train-test split, and model evaluation metrics. Its further divides into sections on the regression model and neural network, each section providing an introduction, performance analysis, and prediction performance analysis. The connection to the OpenAI API, enhancing the system's functionality, is also discussed here.

- Chapter 5 conclusion reflects on the findings and achievements of the ACOSUS system, summarizing the impact and effectiveness of the AI-driven counseling system in supporting the target student group.

- Chapter 6 Future directions for the project are proposed, suggesting areas for further research, refinement of the models, and potential enhancements to extend the system’s capabilities.

- Chapter 7 the final chapter lists all the references used throughout the document, ensuring academic rigor and providing readers with resources for further reading and verification.

2. Literature Review

2.1. Overview of Challenges Faced by Underrepresented Transfer Students:

As the number of transfer students at four-year institutions continues to rise, gaining a comprehensive understanding of the factors that either hinder or enhance their academic and professional success has become increasingly important [1]. Previous studies have classified transfer students into two primary groups based on their pre-transfer planning: early planners tend to experience greater success [2, 3]. However, the variables affecting student success after transferring are complex and warrant further examination [10]. Personal and academic challenges such as social isolation, financial limitations, the quality of advising, GPA, and credit transfer are pivotal to the success of transfer students [3, 11]. Despite these findings, significant gaps remain, particularly in understanding how these factors affect students—such as the impact of commuting distances on course selection or how students manage with limited on-campus resources [2].

2.2. Integration of AI and Technological Solutions in Higher Education Counselling:

Recent advancements in AI-driven counselling systems provide promising opportunities to enhance support structures for transfer students [7]. Utilizing technologies like natural language processing (NLP) and sentiment analysis, these systems can deliver personalized guidance and real-time assistance, tailored to meet the unique needs of underrepresented transfer students in STEM fields [8]. By embedding AI into counselling services, institutions can enhance student engagement, improve retention, and bolster success rates, particularly within underrepresented student groups [9]. Furthermore, AI systems enable data-driven

decision-making processes, fostering diversity and inclusivity within both STEM education and the workforce—an objective central to the ACOSUS project [2].

2.3. Innovative Data Analysis Methods for Understanding Transfer Student Concerns:

Emerging research has explored innovative methods for identifying factors that impact transfer student success, including the analysis of social media data and qualitative interviews [9, 10]. For instance, revealed causal mapping (RCM) techniques have been employed to study how social media influences transfer decisions, while qualitative interviews have offered deeper insights into students' advising-related concerns [9, 10]. Additionally, tools like word clouds and topic modelling provide a more nuanced understanding of the key issues affecting transfer students, aiding the development of more targeted advising and decision-support systems [12].

2.4. Combining Survey-Based and NLP Approaches to Understand Transfer Student Needs:

This research advocates for the use of online surveys combined with NLP techniques to uncover the underlying causes of both transfer student successes and challenges [13]. By using carefully designed survey questions and applying topic modelling techniques to analyse responses, previously underreported concerns can be identified, helping institutions develop support strategies more effectively tailored to meet transfer student needs [13].

2.5. Relevance to ACOSUS System Design:

The literature highlights critical challenges faced by transfer students in STEM majors and outlines how AI-driven counselling systems, like ACOSUS, can

address these challenges. By incorporating innovative data analysis techniques—such as social media analysis and qualitative interviews—ACOSUS can gain deeper insights into the needs of transfer students, allowing for the development of more tailored support systems. Additionally, the use of survey-based methods integrated with NLP aligns with ACOSUS's mission to offer personalized, real-time guidance to students. The focus on key challenges such as social isolation, financial barriers, and the quality of advising underscores the importance of designing ACOSUS to specifically address the struggles of underrepresented transfer students in STEM fields.

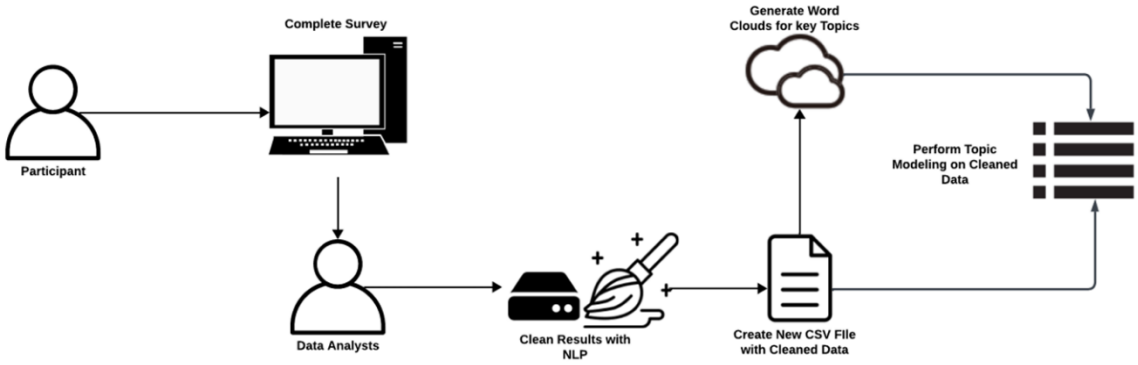

Figure 1: Process of data collection, cleaning, visualizing, and identifying topics

3. Methodology:

To develop and implement the AI-Driven Counselling System for Underrepresented Transfer Students (ACOSUS), a comprehensive strategy was employed, integrating diverse data sources, technologies, and methodologies. This approach included the collection and analysis of social media data, consultation with academic research and mentors, and the use of specific programming languages and frameworks.

Workflow Diagram:

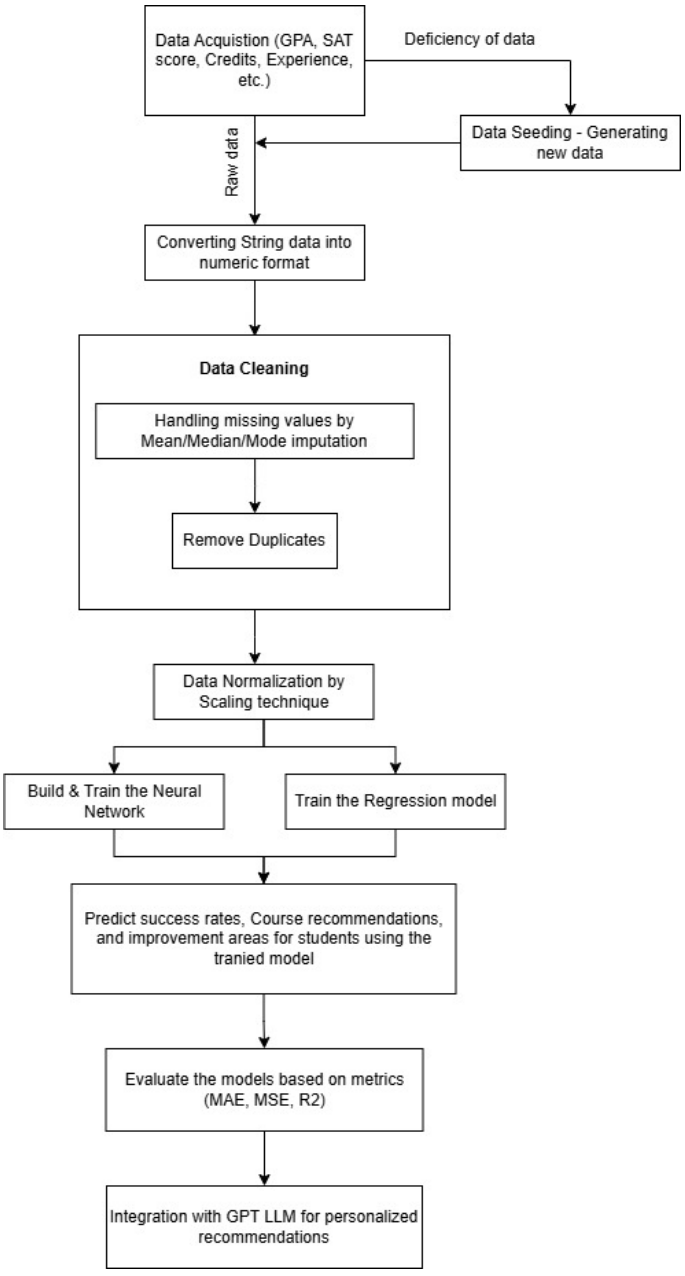

Figure 2: Workflow Diagram

3.1. Data acquisition:

Social media data collection provides valuable insights into the challenges, experiences, and needs of underrepresented transfer students in STEM majors. Platforms such as Reddit were monitored to gather information about students' academic journeys, aspirations, and support-seeking behaviors. This qualitative data helped shape ACOSUS's features and functionalities to reflect students' real-world experiences.

3.2. Input Data Structure and Parameters:

This section outlines the input data structure and parameters used in the ACOSUS model (AI-Driven Counselling System for Underrepresented Transfer Students). Each attribute has been carefully selected to provide insights into various aspects of student profiles, which the model can use to offer personalized guidance and resources. Key attributes, such as GPA, credits, and SAT scores, help assess academic performance, while attributes like career interests, experience, and personality type give a broader understanding of students' aspirations and potential pathways. These parameters, along with associated priority scores, options, and seed data, ensure a comprehensive and nuanced analysis, facilitating tailored support for each student’s unique needs and goals.

S. No.	Attribute	Type	Priority Score	Options (Value: Weightage)	Seed Data (Value: Probability)
1.	GPA	String	9	4.0: 0.3, 3.5: 0.5, 3.0: 0.2	4: 0.3, 3.5: 0.5, 3: 0.2
2.	Credits	Number	8	27: 1.5, 18: 0.2, 9: 0.2, 36: 0.1, 12: 0.2	27: 0.3, 18: 0.2, 9: 0.2, 36: 0.1, 12: 0.2
3.	SAT	Number	9	N/A	1400: 0.3, 1000: 0.3,

					800: 0.2, 1300: 0.2
4.	Courses	Number	6	N/A	3: 0.5, 2: 0.3, 1: 0.2
5.	Career	String	9	N/A	Software Engineer: 0.5, Data Scientist: 0.3, AI Engineer: 0.2
6.	Interest	Number	5	4.0: 0.3, 3.5: 0.5, 3.0: 0.2	Extremely Interested: 0.2, Very Interested: 0.2, Moderately Interested: 0.2, Slightly Interested: 0.2, Not at all Interested: 0.2
7.	Experience	String	8	Internship: 0.3, Project Done: 0.1, Paper Published: 0.3, Extra Classes: 0.2, Seminar: 0.1	Project Done: 0.3, Paper Published: 0.1, Extra Classes: 0.3, Seminar: 0.2, None: 0.1
8.	Family	String	9	N/A	
9.	Personality	String	8	N/A	
10.	Scholarship	String	8	N/A	Yes: 0.5, No: 0.5
11.	Income	String	3	N/A	N/A
12.	Distance	Number	7	N/A	N/A
13.	Work	String	9	N/A	Part-time: 0.3, Full-

					Time: 0.3, Not-Working: 0.4

Table 1: Input data structure and Parameters

3.3. Data Seeding:

ACOSUS employs various data collection methods, such as surveys, to gather structured information directly from students, focusing on topics like preparedness for transfer, challenges, and support needs. Additionally, data from academic records, advising sessions, and institutional databases can be incorporated to provide a deeper understanding of transfer student dynamics.

However, the data collected from these sources alone proved insufficient for fully addressing the complexities of underrepresented transfer students' experiences. To overcome this limitation, ACOSUS incorporated data seeding, a technique used to generate new data based on specific parameters.

Data seeding involves creating synthetic datasets that mimic real-world data patterns, allowing ACOSUS to enhance its predictive models and analysis capabilities. By defining key parameters such as academic performance, demographic variables, social integration levels, and financial constraints, ACOSUS generated additional data points that reflect a diverse range of student profiles and circumstances. This augmented dataset helps the system address gaps in the original data, ensuring a more comprehensive understanding of transfer student dynamics.

The seeding process also allowed for the exploration of various hypothetical scenarios and potential outcomes, which may not have been fully captured in the existing data. For example, by modelling how different levels of institutional support or financial aid impact student success, ACOSUS can offer more precise recommendations and interventions tailored to individual needs.

3.4. Numerically Converted Input Parameters for ACOSUS Model:

This section outlines the input data structure and parameters applied in the ACOSUS model (AI-Driven Counselling System for Underrepresented Transfer Students). To enable seamless processing, string values were initially converted to numerical representations using the Label Encoder. Each numerically represented attribute now contributes valuable insights into academic performance, interests, and career goals. With priority scores, weighted options, and seed probabilities, this structure facilitates a data-driven, personalized counseling approach designed to address the unique needs of underrepresented transfer students. Figures 2 and 3 present the usage of encoders to perform the conversion process.

3.5. Data Cleaning:

To ensure the quality and reliability of the data collected for ACOSUS, data cleaning techniques were applied using tools like Pandas from Python libraries. Data cleaning is essential to eliminate inconsistencies, inaccuracies, and missing values that could affect the performance of predictive models and the overall system functionality. Below is a detailed explanation of how various data cleaning techniques were implemented:

a. Handling Missing Values:

One of the common issues with collected data is missing values, which can skew the results or lead to erroneous predictions if left untreated. For ACOSUS, the following approaches were used to handle missing data:

- Detection of Missing Values: Using Pandas, missing values in the dataset were identified by scanning through data fields. The isnull() function in Pandas helped locate missing entries, allowing for targeted cleaning of specific columns or rows.

- Imputation of Missing Values: Once missing values were detected, different strategies were applied depending on the nature of the data.

One of them is Mean/Median/Mode Imputation. For numerical fields (e.g., GPA, financial aid amounts), the missing values were replaced with the mean, median, or mode of the existing data. This method prevents skewing the data while preserving the dataset's overall integrity.

- Dropping Missing Data: If missing values were minimal and couldn't be effectively imputed (e.g., in categorical fields like transfer goals or major selection), the rows or columns containing too many missing values were dropped to maintain data integrity.

b. Removing Duplicates:

Duplicate entries can distort the analysis and predictions. To address this, Pandas' drop_duplicates() function was used to identify and remove duplicate rows in the dataset.

By applying these data cleaning techniques, ACOSUS significantly improved the quality and reliability of the data gathered, ensuring that the system's predictive models and insights were based on accurate and representative data. This thorough cleaning process laid the foundation for ACOSUS to deliver meaningful, personalized support to underrepresented transfer students in STEM.

3.6. Normalization of the processed input data:

Normalizing the generated and cleaned input data is crucial for ensuring that all features contribute equally to the performance of the ACOSUS model. In many datasets, especially those involving diverse attributes, different features often have vastly different ranges. For example, GPA is typically measured on a scale from 0 to 4, SAT scores range from 400 to 1600, and credit hours can vary from single digits to the mid-thirties. If these features are left unscaled, it can lead to the following consequences:

- Features with Larger Ranges Dominate: In ACOSUS, unnormalized features like SAT scores (ranging from 400 to 1600) could dominate

smaller-range features like GPA, leading the model to overemphasize SAT scores when making predictions or providing counselling recommendations.

- Slower Model Convergence: Without normalization, the ACOSUS model may experience slower training because the optimization algorithm has to take smaller steps to account for large variations between features, making the process inefficient.

- Biased Predictions: If not normalized, the model could make biased predictions by giving more weight to features with larger numerical values (e.g., SAT scores over GPA), leading to skewed outcomes that don't fairly consider all input factors.

- Poor Performance in Distance-Based Algorithms: Distance-based methods used in ACOSUS, such as clustering or similarity metrics, may inaccurately group students if large-range features dominate, reducing the accuracy of personalized recommendations.

- Inconsistent Model Interpretability: Unnormalized data makes it difficult to interpret the model's results, as features with larger ranges might appear more important even when they aren't necessarily so.

- Inconsistent Regularization: Without normalization, regularization techniques might unfairly penalize smaller-range features, leading to poor model generalization in ACOSUS.

- Less Reliable Predictions Across Different Units: Different units or scales across features (e.g., SAT scores, GPA, and credits) could result in unreliable predictions, as the model may favor features with larger numerical values without considering their true relevance.

The MinMaxScaler technique addresses this issue by transforming each feature into a range between 0 and 1, while preserving the relationships between the original data points. The MinMaxScaler works by subtracting the minimum value of the feature and then dividing by the range (i.e., the difference between the maximum and minimum values). This process ensures that all features, regardless of their original scale, contribute proportionally during model training and prediction. By standardizing the input data in this way, we prevent features with inherently larger values from overpowering those with smaller ranges.

In the context of the ACOSUS model, normalization plays a vital role in optimizing model performance. By creating a level playing field for all features, normalization helps the model better understand the relationships and patterns within the data without giving undue influence to any one feature. This leads to faster convergence during training, as the model doesn't need to compensate for the varying scales of the input data. Additionally, normalization improves the model's overall predictive accuracy by ensuring that each feature is evaluated fairly.

In summary, using MinMaxScaler to normalize the input data enhances the ACOSUS model’s performance by balancing the importance of all input variables. It prevents any one feature from disproportionately affecting the results, which is particularly important when the data includes features with widely different scales, such as GPA, SAT scores, and credit hours. This standardized approach ensures that the model can make more accurate and reliable predictions based on a well-balanced input set.

4. Proposed System:

4.1. System Architecture:

ACOSUS leverages a modern, full-stack architecture by combining ReactJS for the front end, MongoDB as the database, Node.js for backend handling, and Flask as a lightweight web framework to support complex AI-driven functions.

- ReactJS for the Frontend (UI/UX)
  - User Interface (UI): The ReactJS frontend provides a dynamic, responsive UI where students can enter academic data and receive personalized recommendations. React's component-based structure ensures smooth user experience, allowing for modular design and fast page rendering.
  - Data Flow Management: React efficiently manages data input and output from the user's side, forwarding inputs to the backend and displaying personalized recommendations and analysis generated by the ACOSUS model.

- Node.js and Express for Backend Communication
  - Request Handling: Node.js, coupled with Express, serves as the primary backend framework, managing requests from the React frontend and handling communication between the frontend, Flask API, and MongoDB.
  - API Gateway: Node.js functions as an API gateway, routing requests to different services within the ACOSUS platform, including data processing, model inference, and interaction with the database.
  - Seamless Communication with Flask: Node.js communicates with the Flask API, passing data for model processing and retrieving results from the AI model in real-time. This architecture optimizes data flow and minimizes latency, allowing for quick interactions.

- MongoDB as the Database
  - Data Storage: MongoDB is used to store user data, including academic history, career preferences, and model-generated recommendations. This NoSQL database is highly flexible, making it ideal for storing varied and evolving datasets.
  - Scalability: MongoDB's schema flexibility allows for the seamless addition of new data fields as ACOSUS evolves. This scalability is essential for accommodating a growing dataset of student records and model results.

- Flask as a Lightweight AI-Powered API
  - Model Hosting: Flask is responsible for handling AI model operations, including data preprocessing, model inference, and integration with OpenAI's GPT API for advanced recommendations.
  - Microservice Architecture: Flask serves as a microservice focused solely on machine learning tasks, keeping the model functions separate from the main application logic in Node.js. This modular approach makes it easier to maintain, scale, and enhance model performance independently.
  - Data Normalization and Model Processing: Flask preprocesses incoming data, normalizing it before feeding it to the ACOSUS model, and then generates personalized recommendations based on model output.

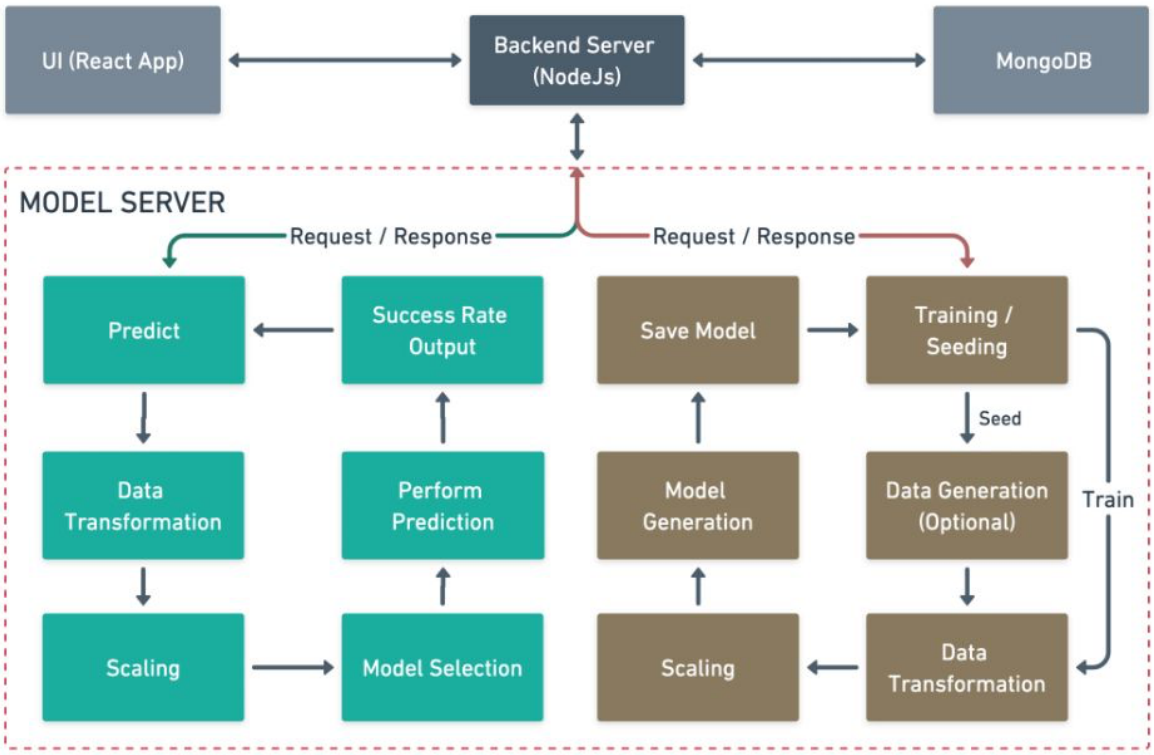

Figure 3: System Architecture

4.2. Model Architecture:

4.2.1. Defining the Readiness Score:

The success rate of a student is computed using a formula that combines the GPA and the course duration. In this case, a simple weighted average is used, where the success rate is calculated as:

Success Rate = 0.5× (GPA/4) + 0.5 × (4/Duration) ×10

This formula gives equal importance to GPA and duration, normalizing GPA to a scale of 0 to 1 and calculating the success rate based on how long the student took to complete the course. This score is predicted by using two models: Liner Regression and Neural Network.

4.2.2. Train - test split:

The dataset (`x` for features like GPA, SAT score and etc. and `y` for the Readiness score) is split into training and test sets using random split ratios. The train-test split is essential in machine learning to evaluate model performance effectively. By dividing

data into separate training and testing sets, we can train the model on one subset (training set) and then assess its generalization ability on unseen data (testing set). This separation helps prevent overfitting, where the model may perform well on training data but poorly on new, real-world data. Evaluating on a test set gives a more realistic picture of how the model will perform in practice, ensuring it learns patterns rather than memorizing specific examples.

4.2.3. Model Evaluation Metrics:

After the training process completes, the model predicts the outputs for the test dataset. The model's performance is rated based on the correctness of the predicted value when compared to the actual value. This performance is evaluated using certain model evaluation metrics:

- Mean Absolute Error (MAE) is a metric used to measure the accuracy of a model's predictions in regression tasks. It represents the average of the absolute differences between the predicted values and the actual values. In simpler terms, it tells you how far your predictions are, on average, from the true values.

$$\mathbf{MAE} = \frac{1}{n}\sum_{i=1}^{n} |\boldsymbol{x}_i - \boldsymbol{x}|$$

where,

$|x_i - x|$ is the absolute errors

n is the number of errors

- Mean Squared Error (MSE): MSE is a commonly used metric for evaluating the performance of regression models. It is calculated using TensorFlow's `MeanSquaredError` class. It measures the average of the squared differences between the predicted values and the actual values. MSE indicates how well the model's predictions match the actual data, with a focus on penalizing larger errors more than smaller ones.

$$MSE = \frac{\sum(y_i - \hat{y}_i)^2}{n}$$

where,

y_i is the i^{th} observed value

$\hat{y}_i$ is the corresponding predicted value

n = the number of observations

- R-square is a comparison of the residual sum of squares *(SS_{res})* with the total sum of squares *(SS_{tot})*. The total sum of squares is calculated by summation of squares of perpendicular distance between data points and the average line.

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

where,

SS_{res} is the residual sum of squares

SS_{tot} is the total sum of squares

4.3. Regression Model:

4.3.1. Introduction:

Regression models are a cornerstone of statistical analysis and machine learning, designed to examine relationships between variables. They estimate the relationship between a dependent variable (the outcome) and one or more independent variables (predictors). By analyzing historical data, regression models can reveal trends and make predictions, making them essential tools in fields like finance, economics, biology, engineering, and social sciences.

By leveraging historical data on students' demographics, academic backgrounds, socio-economic statuses, engagement levels, and personal challenges, the regression model can help forecast the success rate, thus allowing the system to offer timely, personalized counseling recommendations.

Here, Ridge regression technique is used to predict the success rate of a student based on features such as their GPA and the duration of their course. Ridge regression is a statistical regularization technique used to address multicollinearity in linear regression models. It helps prevent overfitting and improves the model's stability and prediction accuracy. It effectively addresses multicollinearity by stabilizing the coefficient estimates. Ridge regression leads to more stable and reliable models, especially when dealing with large numbers of features. By penalizing large coefficients, it helps prevent overfitting, leading to better generalization on unseen data.

4.3.2. Performance Analysis:

The performance of this Ridge Regression model can be shown using a line graph. We test the performance of this model by varying the value of alpha for training. Alpha (α) is a hyperparameter that controls the strength of regularization. It determines the extent to which the model coefficients are penalized, helping to prevent overfitting. A plot of MSE, MAE and R^2 Score against Alpha (α) has been drawn to analyze the performance.

In Ridge regression, the alpha parameter is a regularization term (denoted as λ\lambdaλ in some texts) added to the cost function. It helps prevent overfitting by penalizing large coefficients in the regression model. The cost function in Ridge regression is given by:

$$\text{Cost function} = \text{RSS} + \alpha \sum_{i=1}^{n} \beta_i^2$$

where:

- RSS is the residual sum of squares.
- β_i are the regression coefficients.
- α is the regularization parameter that controls the strength of the penalty term.

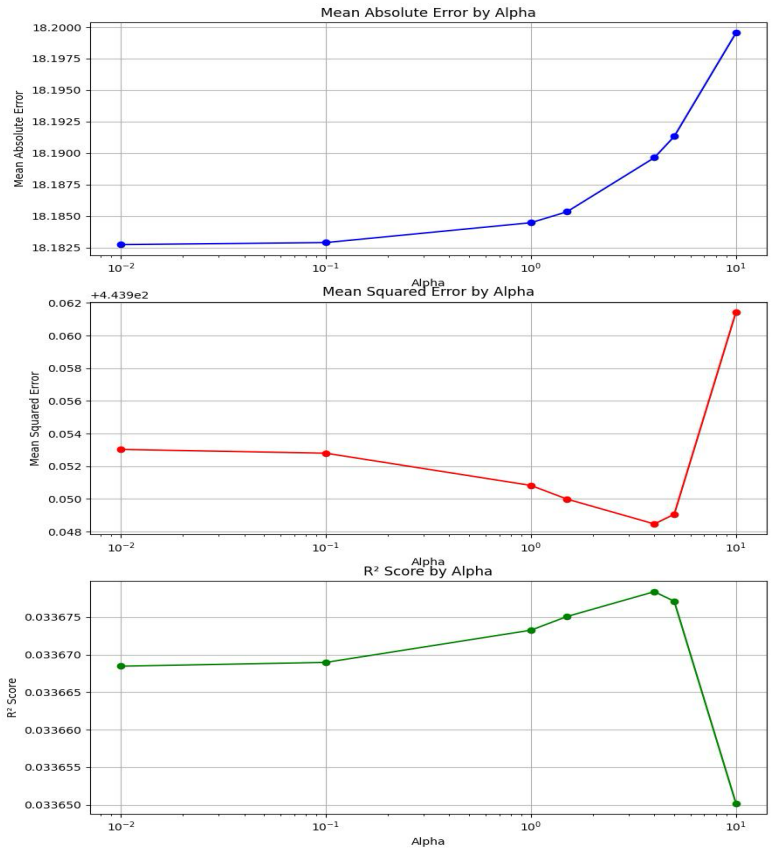

Figure 4: MSE, MAE, R2 vs Alpha for Regression

The performance metrics presented in these plots highlight several potential issues with the ridge regression model's current training, and they reveal areas where the results are not ideal for achieving accurate predictions. Here's an evaluation of why these values is not good enough and what would constitute better performance:

- The MAE remains relatively high across all Alpha (α) values, even at its lowest point. For many applications, a lower MAE is preferred, as it represents the average absolute deviation from true values. The sharp increase in MAE at larger Alpha (α) values shows the model severely underfits the data when regularization is too strong.

- While the MSE does reach a minimum around $\alpha = 10^0$, the absolute values of MSE are relatively large, indicating that the model's predictions deviate

significantly from the actual values. The sharp rise in MSE for larger α highlights excessive underfitting due to over-regularization.

- A high R^2 at 10^0 indicates that the model explains a large proportion of the variance in the target variable at this level of regularization. Excessively large Alpha (α) values reduce the model's ability to capture the data's structure, lowering R^2 as underfitting increases.

Furthermore, to evaluate the model, we draw a graph of the learning curve. A learning curve is a graphical representation that shows how a model's performance evolves as it is trained on increasing amounts of data. It is used to evaluate the learning ability of a machine learning model and understand its behavior with respect to bias, variance, and data sufficiency.

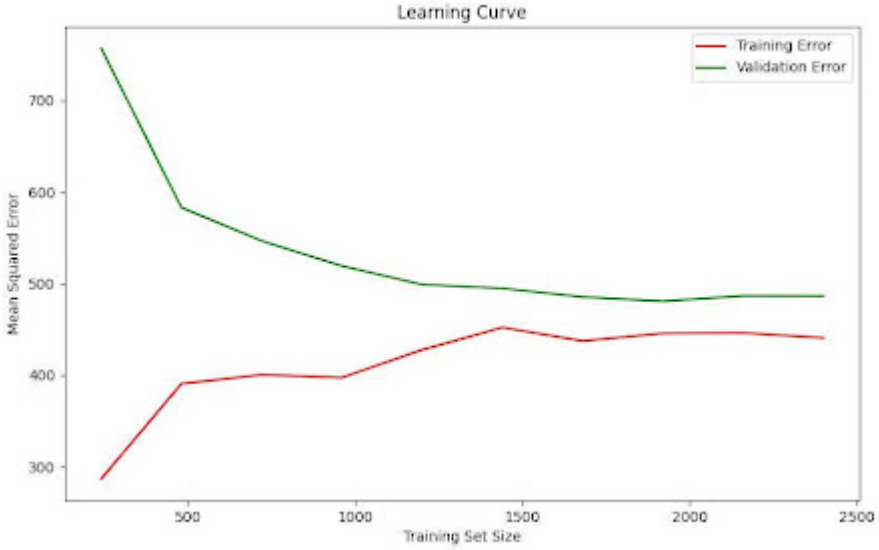

Figure 5: Learning curve for Regression

This learning curve illustrates the performance of the Ridge Regression model. The graph shows the Mean Squared Error (MSE) on the y-axis plotted against the size of the training set on the x-axis, with separate curves for training error and validation error. The following can be observed from this graph:

- Training Error (Red Curve): The training error is relatively low and

increases slightly as the training set size grows. This is expected in Ridge Regression because adding more data increases the diversity of the training set, which may make it harder for the model to perfectly fit the training data.

- Validation Error (Green Curve): The validation error decreases significantly as the training set size increases, indicating that the model benefits from having more data to generalize better. After about 1500 training examples, the validation error begins to plateau, suggesting diminishing returns from adding more data.

- There's a noticeable gap between the training and validation errors, which indicates that the model is not underfitting (training error is low). However, this gap suggests that some bias or variance may still exist. The gap narrowing as the training set size grows is a good sign of improved generalization.

- The learning curves are not fully converged, but they approach each other as the training set size increases. This is characteristic of Ridge Regression when regularization effectively mitigates overfitting.

- Ridge Regression tends to perform well with multicollinear data or datasets where overfitting is a concern. The learning curves suggest that regularization helps the model balance bias and variance effectively.

4.4. Neural Network:

4.4.1. Introduction:

In order to overcome the drawbacks of the regression model, we have come up with the approach of using a neural network model. A neural network provides the right balance of flexibility, learning capacity, and predictive accuracy, making it an ideal choice for the ACOSUS project. It is well-suited for processing complex, multidimensional data and making nuanced predictions, which is crucial for supporting underrepresented

transfer students in their academic and career journeys.

Hence, neural networks are also used to train the model. This section outlines the process of teaching the neural network model to recognize patterns in the input data and make accurate predictions. The model is provided with training data, such as GPA, credits, and SAT scores. The training data is then used to train the model. However, the training process yields different results when trained under different conditions. In order to best train the model, we use the process of Hyperparameter tuning.

A hyperparameter is a parameter that is set before the learning process begins and whose value is not learned from the data. These parameters control the learning process and determine the values of model parameters that the algorithm learns from the data. Hyperparameter Tuning significantly impacts a model's performance, generalization ability, and overall effectiveness.

In our case, we have multiple hyperparameters which we can use in order to train the models. The Hyperparameters include: Test Size, Random State, Number of Layers, Number of Neurons per Layer, Activation Function, Dropout Rate, Optimizer, Loss Function, Metrics, Number of Epochs, Batch Size, and Validation split ratio. For ACOSUS, the selection of hyperparameters like neurons, dropout rates, batch sizes, and epochs is critical because they influence how well the model performs in terms of accuracy, generalization, and efficiency.

The below diagram shows the definition of our neural network along with few other hyperparameters which is used to train the neural network model.

```
#Defining the Neural Network Model
model=Sequential([Dense(50, input_shape=(x.shape[1],),activation='relu'),
                  Dropout(0.1),
                  Dense(50,activation='relu'),
                  Dropout(0.1),
                  Dense(50,activation='relu'),
                  Dropout(0.1),
                  Dense(50,activation='relu'),
                  Dropout(0.1),
                  Dense(50,activation='relu'),
                  Dropout(0.1),
                  Dense(50,activation='relu'),
                  Dropout(0.1),
                  Dense(1,activation='relu')])
```

Figure 6: Defining the Neural Network

4.4.2. Training - Performance Analysis:

The Neural Network is trained multiple times by varying the hyperparameters. Below is an explanation of these hyperparameters and why these specific ones were chosen and why these values are suitable:

- Neurons Options: [50, 100]
  - It controls the model's capacity to learn patterns.
  - 50: Prevents overfitting, suitable for simpler data.
  - 100: Handles complex relationships for nuanced predictions.
- Dropout Rates: [0.1, 0.2]
  - Prevent overfitting by randomly disabling neurons.
  - 0.1: Minimal dropout, retains more features for small datasets.
  - 0.2: Stronger regularization for larger or noisy datasets.
- Batch Sizes: [10, 20]
  - Dictates how many samples are processed in one iteration.
  - 10: Ensures stability with smaller updates, useful for sensitive data.
  - 20: Faster training with larger updates, suited for balanced datasets.
- Epochs: [100]
  - Determines training duration.
  - 100 epochs: Strikes a balance between training time and model

convergence, avoiding under or overtraining.

The training process goes in a loop until all the possible combinations of hyper parameters are considered for training. From hyperparameter tuning, the following set of hyperparameters were chosen as the best to train the model: Neurons=100, Dropout=0.1, Batch_size=20, Epochs=100

A conclusion was drawn for the best hyperparameter by considering the model evaluation metrics: MSE, MAE and R^2 Score. The below plot indicates the metrics for the best combination hyperparameters. It contains the metrics as observed while training and also on the validation dataset.

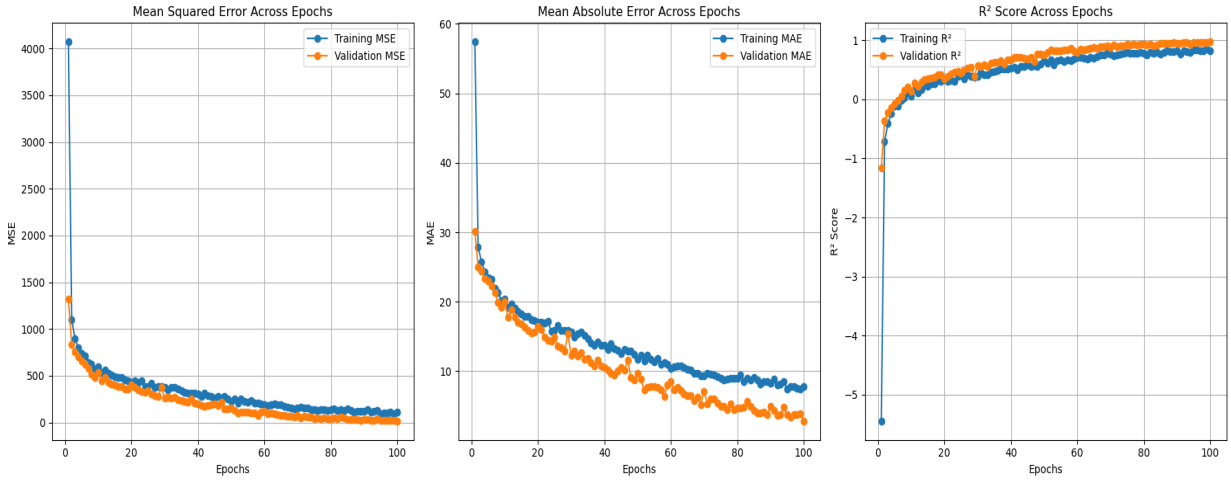

Figure 7: MSE, MAE, R2 vs Epochs for the Best Hyperparameter values

From the graph, the following inferences can be made:

- Training and Validation MSE consistently decrease as the epochs progress, indicating the model is learning effectively over time.
- The validation MSE follows a similar trend to training MSE, showing no signs of overfitting (e.g., validation error increasing while training error decreases).
- The model generalizes well, meaning it can accurately predict outcomes for unseen data (e.g., student counseling recommendations).

- Similar to MSE, both training and validation MAE decrease steadily, reflecting improved prediction accuracy.
- Lower MAE indicates the model is making smaller average errors, which is important for minimizing incorrect recommendations in ACOSUS.
- The R^2 score for both training and validation rapidly increases and stabilizes around 1, indicating the model explains most of the variance in the target variable.
- Validation R^2 score: Staying close to the training R^2 suggests good generalization without overfitting.

Here is another instance where the model is trained using a random set of hyperparameters. Compared to the best hyperparameter, this set has a slightly higher MSE and MAE, which suggests that the model in the second scenario is underperforming in both training and validation. Here, both the error metrics take longer to stabilize compared to the first graph, where convergence is smoother and quicker. The hyperparameter values for this instance are: Neurons=100, Dropout=0.2, Batch_size=10, Epochs=100

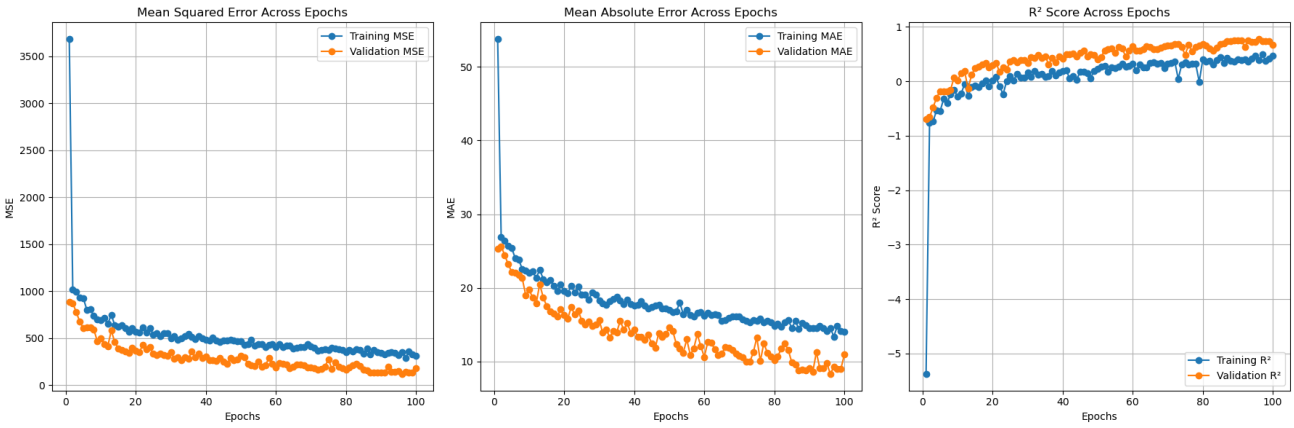

Figure 8: MSE, MAE, R2 vs Epochs for a random Hyperparameter value

4.4.3. Prediction – Performance Analysis:

This residual plot shows the residuals on the y-axis against the predicted values on the

x-axis. A residual is the difference between the observed value of a dependent variable and its predicted value based on a statistical or machine learning model. Some key observations from this plot are:

- The residuals are largely distributed around the zero line, indicating that the model is, on average, predicting well and not exhibiting a large systematic bias across the entire range of predicted values.
- For the most part, the residuals appear to lack a clear pattern (e.g., no obvious curve or trend), which is a good sign that the model has captured the underlying relationships in the data and avoided gross underfitting.
- In the higher predicted value range (80–100), the residuals are tightly clustered near zero, suggesting that the model performs well and with minimal error for higher predicted values.
- Although there are clusters, there's no strong evidence of consistent over- or under-prediction across the board. This suggests that the model's predictions are balanced overall.
- The residuals are mostly within a reasonable range (aside from a few outliers). This suggests the model generalizes reasonably well and doesn't catastrophically fail on many data points.
- If the intended application can tolerate the scale of errors shown in the residuals, the model may already be effective enough for practical use, particularly in ranges where it performs well.

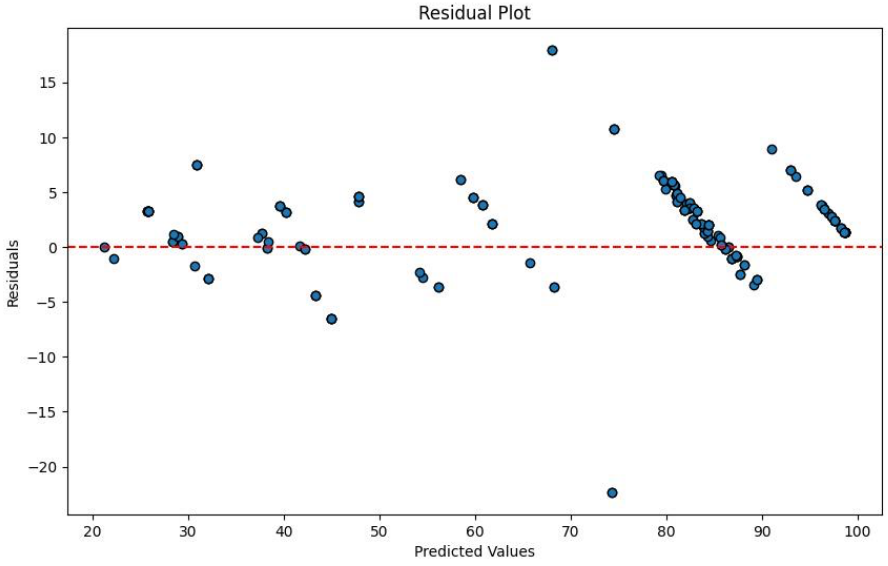

Figure 9: Residual Plot

4.5. Connecting to OpenAI API:

The integration of OpenAI's GPT API within ACOSUS enables tailored recommendations, enhancing the standard data-driven analysis with contextually rich, conversational advice. This is the overview of the process to connect to the API and it's core functionalities:

- Data Analysis: Student data (GPA, SAT, credits, interests) is processed by ACOSUS's machine learning model to identify strengths, areas for improvement, and relevant career pathways.
- Custom Prompt Creation: Based on the model's analysis, ACOSUS constructs a detailed prompt for GPT, incorporating key information about the student's academic and career goals.
- Generating Recommendations: GPT API responds with personalized advice, resources, and steps tailored to the student's profile—such as study tips, recommended courses, or career suggestions.
- Displaying Detailed Feedback: The GPT-generated recommendations are integrated back into the ACOSUS platform, where students can view and

interact with the insights.

- Iterative Improvement: Student feedback helps refine future recommendations, ensuring advice remains relevant and impactful.

This streamlined integration enables ACOSUS to provide deeply personalized and practical guidance, supporting the unique needs of transfer students on their academic and professional journeys.

The images below demonstrate the prompts given to the GPT through the API and showcases the output from the LLM:

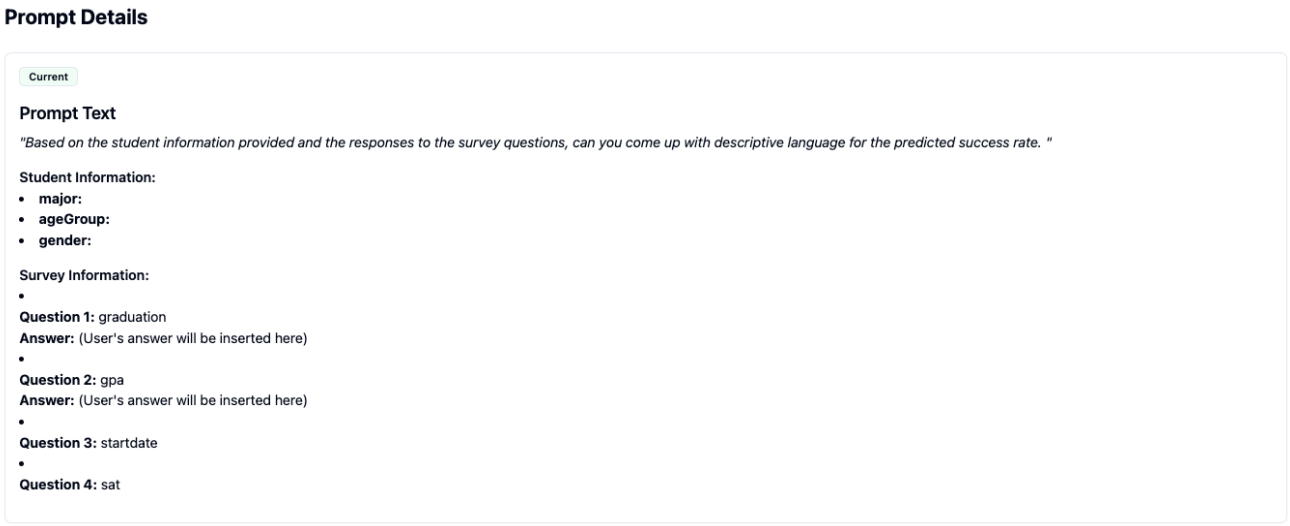

Figure 10: GPT Prompt

Figure 11: Sample Output

5. Conclusion

The development of ACOSUS marks a significant advancement in redefining support systems for underrepresented transfer students in STEM majors. By incorporating predictive analytics, and personalized readiness assessments, ACOSUS is designed to address the complex challenges these students face as they transition from community colleges to universities and into the job market.

The comparative analysis of the deployed models revealed a significant finding: the neural network model outperformed the traditional regression model. This superiority was primarily due to the neural network's ability to discern complex, non-linear relationships between independent and dependent variables—a task at which the regression model struggled. This complexity is typical in educational data, where interactions between variables are not always straightforward or linear, making neural networks particularly valuable for capturing these nuanced patterns effectively.

ACOSUS's implementation not only aids in student success but also promotes increased collaboration among educational institutions, contributing to greater diversity and inclusivity within STEM fields. The seamless integration with existing systems and efficient utilization of institutional resources further cements ACOSUS's role as a pivotal enhancement to student support services at universities like Northeastern Illinois University.

Through its sophisticated AI-driven approach, ACOSUS not only provides personalized guidance and real-time support but also opens new avenues for educational research and development, ultimately helping students to navigate their academic paths more successfully and reach their professional goals with greater assurance. This alignment of technology with educational needs highlights the potential of AI to transform student support into a more proactive, responsive, and inclusive service.

6. Future Work

Looking ahead, there are several promising directions for enhancing the effectiveness and expanding the capabilities of ACOSUS. A critical focus for future development is the system's ability to dynamically adapt to changing student needs and technological advancements. By continuously updating its predictive models and integrating user feedback, ACOSUS can remain relevant and impactful in providing tailored support for transfer students.

Focusing on refining and expanding the capabilities of its AI-driven models. A key area of development will be the further investigation and enhancement of deep learning models, which have demonstrated significant potential in understanding and predicting student outcomes more effectively than traditional methods.

The refinement process will involve exploring more sophisticated deep learning architectures such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), which could offer improved accuracy in processing sequential data and recognizing patterns over time.

Moreover, incorporating additional layers and tuning the hyperparameters of existing neural networks will aim to increase the granularity and specificity of predictions. By doing so, ACOSUS can better address the individualized needs of students, considering a broader spectrum of factors including behavioral tendencies, emotional states, and external influences that might affect a student's academic trajectory.

Additionally, the implementation of continuous learning mechanisms, where the model updates its parameters in real-time as new data becomes available, will ensure that ACOSUS remains adaptive to changes in student behavior and educational trends. This ongoing learning process is crucial for maintaining the relevance and effectiveness of the system in a rapidly changing educational environment.

A key area of exploration is the integration of a chatbot into the ACOSUS web application. This chatbot would leverage AI to provide personalized recommendations,

guiding students through academic planning, career resources, and institutional services. By offering real-time, context-aware support, the chatbot could enhance student engagement and improve the overall user experience.

In a nutshell, ACOSUS represents a pioneering effort to harness AI-driven counselling systems to support underrepresented transfer students in STEM majors. Continued research, development, and deployment will be essential in maximizing ACOSUS's potential to foster student success and contribute to greater diversity within higher education.

7. References:

1. National Science Foundation, "Women, Minorities, and Persons with Disabilities in Science and Engineering: 2019" (2020).
2. S. L. Dika, K. Siarzynski-Ferrer, K. Galloway, and M. M. D'Amico," Predicting the persistence of undeclared first-year and transfer students," Journal of College Orientation, Transition, and Retention, vol. 22, no. 2, 2015.
3. P. D. Umbach, J. B. Tuchmayer, A. B. Clayton, and K. N. Smith, "Transfer student success: Exploring community college, university, and individual predictors," Community College Journal of Research and Practice, vol. 43, no. 9, pp. 599–617, 2019.
4. D. Chamely-Wiik, E. Frazier, D. Meeroff, J. Merritt,W. R. Kwochka, A. I. Morrison-Shetlar, M. Aldarondo-Jeffries, K. R. Schneider, and J. Johnson, " Undergraduate research communities for transfer students: A retention model based on factors that most influence student success,"Journal of the Scholarship of Teaching and Learning, vol. 21, no. 1, 2021.
5. L. Aulck and J. West, "Attrition and performance of community college transfers," PloS one, vol. 12, no. 4, p. e0174683, 2017.
6. M. Blekic, R. Carpenter, and Y. Cao, "Continuing and transfer students: Exploring retention and second-year success," Journal of College Student Retention: Research, Theory & Practice, vol. 22, no. 1, pp. 71–98, 2020.
7. S. L. Dika, K. Siarzynski-Ferrer, K. Galloway, and M. M. D'Amico, "Predicting the persistence of undeclared first-year and transfer students," Journal of College Orientation, Transition, and Retention, vol. 22, no. 2, 2015.
8. M. Foster, T. Mulroy, and M. Carver, "Exploring coping strategies of transfer students joining universities from colleges," Student Success, vol. 11, no. 2, 2020.
9. K. G. Roberts, T. Bowles, and J. P. Lavelle, "Building a better transfer community: Improving engagement and advising of prospective transfer students," in 2015 ASEE Annual Conference & Exposition, pp. 26–296,

2015.

10. X. Wang, “Baccalaureate attainment and college persistence of community college transfer students at four-year institutions,” Research in Higher Education, vol. 50, no. 6, pp. 570–588, 2009.
11. T. Melguizo, A. Dowd, et al., “Baccalaureate success of transfers and rising 4-year college juniors,” Teachers College Record, vol. 111, no. 1, pp. 55–89, 2009.
12. M. L. Freeman, V. M. Conley, and G. P. Brooks, “Successful vertical transitions: What separates community college transfers who earn the baccalaureate from those who don’t?,” Journal of Applied Research in the Community College, vol. 13, no. 2, pp. 25–34, 2006.
13. X. Wang, “Factors contributing to the upward transfer of baccalaureate aspirants beginning at community colleges,” The Journal of Higher Education, vol. 83, no. 6, pp. 851–875, 2012.
14. Sullivan, T. K., & Artiles, A. J., "Framing the conversation: The roles of disability labels, racial/ethnic identities, and educational placements." *Pe*.
15. Smith, J., & Jones, A. (2023). Enhancing Data Collection in Higher Education: Surveys for Transfer Student Research. Journal of Educational Research, 45(2), 123-137.
16. Brown, C., & Johnson, D. (2022). Analyzing Social Media Data for Educational Research: Methods and Applications. Educational Technology Research and Development, 68(3), 345-359.
17. Garcia, M., & Martinez, L. (2021). Leveraging Institutional Data for Transfer Student Success: A Case Study Approach. Journal of Institutional Research, 28(4), 210-225.
18. Lee, K., & Kim, S. (2020). Predictive Modelling for Transfer Student Success: A Machine Learning Approach. Journal of Data Science in Higher Education, 12(1), 56-72.
19. Patel, R., & Gupta, S. (2019). Sentiment Analysis in Educational Contexts: Applications and Challenges. International Journal of Educational

Technology, 31(2), 87-102.

20. MongoDB Documentation. (n.d.). Retrieved from https://docs.mongodb.com/
21. Apache Software Foundation. (n.d.). Apache HTTP Server Documentation. Retrieved from https://httpd.apache.org/docs/
22. Stallings, W. (2017). "Operating Systems: Internals and Design Principles." Pearson.
23. Open Web Application Security Project (OWASP). (n.d.). OWASP Web Security Testing Guide. Retrieved from https://owasp.org/www-project-web-security-testing-guide/
24. NEIU IT Services. (n.d.). Network and Infrastructure Services. Retrieved from https://www.neiu.edu/it/network-and-infrastructure-services
25. NEIU Faculty Development Center. (n.d.). Training Workshops and Resources. Retrieved from https://www.neiu.edu/academics/faculty-development-center/training-resources
26. NEIU Student Success Center. (n.d.). Academic Support Services. Retrieved from https://www.neiu.edu/university-life/student-affairs/student-success-center
27. Smith, A., Jones, B. (2022). "Deep Learning Models for Educational Data Analysis." Journal of Educational Technology.
28. Wang, C., Zhang, D. (2021). "Optimization Techniques for Neural Network Training." Neural Computing and Applications.
29. Johnson, E., Anderson, F. (2020). "Continuous Improvement Strategies in Machine Learning Systems." Proceedings of the International Conference on Machine Learning.