EZO Project Report

ISY5001 Intelligent Reasoning Systems

Group Members:
Li Yiyao
Zhu Xuanya
Liang Zhu
Zhang Qiyuan
Shen Kaiyuan

Catalogue

1. EXECUTIVE SUMMARY

With the advancement of globalization, international exchanges have become more frequent, making studying abroad an effective way to broaden horizons and enhance international competitiveness. More and more students are inclined to pursue their education overseas. In 2022, the number of overseas Chinese students reached up to 830,500.

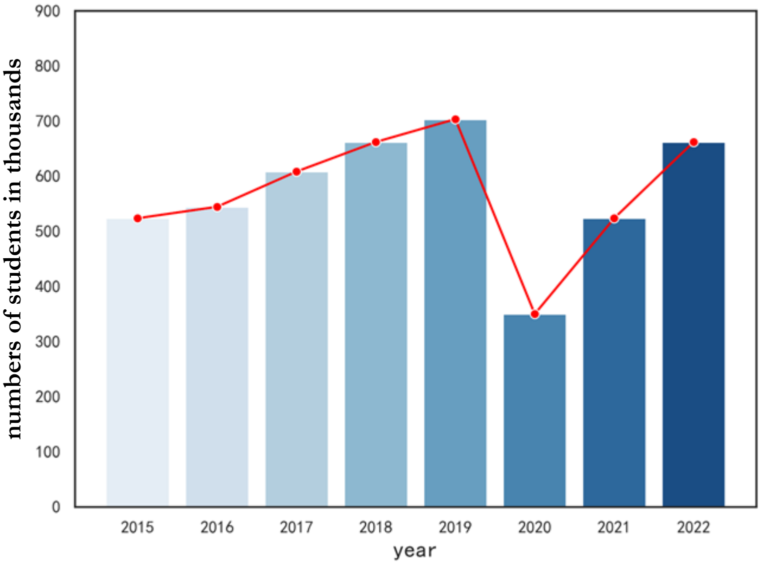


Figure 1: numbers of Chinese students studying abroad

To improve the possibility of getting into a dream school, it’s necessary to get familiar with the relevant information of the target school and major in advance, in conjunction with knowing which school and major are suitable for their own background.

Our project group has co-developed an intelligent system oriented towards students who need consulting on studying abroad. To begin with, we used Dialogflow, LlamaIndex and GPT-3.5 API to make a chatbot, which can answer users’ relevant questions. Users can inquire about various information on schools and majors, including school profiles, professional Settings, etc. This will provide users with a comprehensive view of schools and majors and the ability to make horizontal comparisons.

Moreover, the website provides a predictive model based on the user's academic background information to predict the user's probability of being admitted to a certain major of a certain overseas university. Given the user’s academic performance, language performance, scientific research experience, internship experience and other information, the website will generate an admission probability assessment for users based on their comprehensive academic background. This will help users evaluate and plan their study abroad applications more objectively.

Besides consulting service, our website also provides users with professional career personality testing tools, such as MBTI test and Holland career test. By completing the tests, users can find what majors they are suitable for.

Under tacit cooperation, our team had an efficient development on this project. By providing information about overseas universities and majors, as well as probability prediction for admission, our website will become an important reference and auxiliary tool for students in the application process. We hope that with our website, students who wanna further their study abroad will be able to apply to the most suitable university and major.

2. PROBLEM DESCRIPTION

Before choosing the target universities and majors to apply for, it's essential for applicants to know relevant information as well as having a proper evaluation of themselves. There are two conventional ways to collect information as well as choosing their target school and major. The most common way is to consult an offline agency, which provides applicants with the service of face-to-face consulting. However, it's time-consuming and money-consuming. So nowadays, more and more people are inclined to choose another way: online consulting. As for the mainstream websites for overseas study consulting, the main service provided is manual consulting, which cannot ensure timeliness. Heavy advertising and Lack of interaction also greatly decrease the user experience.

2.1 BUSINESS OBJECTIVES

Our website mainly provides three services: chatbot for overseas study consulting, probability prediction for admission and professional career personality testing.

1. **Chatbot for overseas study consulting**: It has the advantage of sufficient definition of intents, which includes graduate programs from world-renowned universities. It also applies sufficient NER technique, thus being able to accurately return intent to various questions asked by the users.

2. **Probability prediction for admission**: The most important function for users to evaluate their own background and decide which major of which university to apply for. The algorithm of LightGBM and Logistic Regression provide a strong guarantee for the prediction model.

3. **Professional career personality testing**: Enables users to find suitable majors more easily, which is used as a reference.

Applying for overseas universities is not an easy task, the process of which is really complicated. Our project aims at easing the application for studying abroad. The website we've built is equipped with a clear UI and refined functions, which is user-friendly and practical.

2.2 TECHNIQUE OBJECTIVES

1. **Achieve Chatbot Response Time Within 5 Seconds**: Ensure that the chatbot provides responses to user queries within a time frame of 5 seconds or less, optimizing user experience for swift interactions.

2. **Enable Scalable Contextual Memory Storage within Token Limits:** Implement a robust system for storing and retrieving contextual information to facilitate seamless and uninterrupted conversations, while adhering to token limits to accommodate large-scale usage.

3. **Maintain Intent Recognition Accuracy of Not Less Than 95%**: Attain a high level of accuracy in identifying user intents, with a target threshold of no less than 95%, enhancing the chatbot's ability to understand and address user queries accurately.

4. **Handle High Concurrent User Load:** Design the system architecture to efficiently manage high levels of concurrent user interactions, ensuring smooth operation even during peak usage periods.

3. KNOWLEDGE MODELING

3.1 KNOWLEDGE IDENTIFICATION

Knowledge identification sets the groundwork for the next stage encompassing knowledge specification. Information sources that are deemed to be useful are identified in preparation of knowledge acquisition. In the context of building EZO system, several main data sources have been identified and are documented in ***Table 1.***

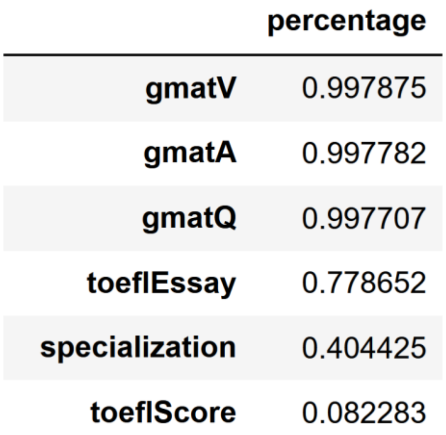
source of information	Insights from information source	Knowledge acquisition technique
university and major information	university general introduction, major information which includes major general description, fees, entry criteria and so on	Web scrapping to obtain publicly available/documented information(https://www.studyabroad.sg/)
QS university rank information	qs top 1000 university names	Web scrapping to obtain publicly available/documented information(https://www.topuniversities.com/university-rankings/world-university-rankings)
Postgraduate application admission data	attributes that determine whether a student is admitted to a graduate program, including major, toefl score, gpa, undergraduate university and so on. the admission flag(successful admission is marked as 1, failure is marked as 0)	Elicitation of tacit knowledge through downloading related data from Kaggle website

Table 1: main data sources

3.2 KNOWLEDGE SPECIFICATION

After acquiring knowledge, it is essential to extract more specific knowledge from unstructured data. Some attributes may be rendered meaningless due to a high incidence of missing values. Additionally, certain attributes, such as usernames, do not have a significant impact on the final prediction results. Consequently, it is imperative to undertake data preprocessing in order to attain more specific and relevant knowledge.

First and foremost, the proportions of missing values for all attributes are computed, and these proportions are then arranged in descending order from the highest to the lowest.



	percentage
gmatV	0.997875
gmatA	0.997782
gmatQ	0.997707
toeflEssay	0.778652
specialization	0.404425
toeflScore	0.082283

Figure 2: the proportions of missing values

gmatV, gamtA, gmatQ, toeflEssay, specialization were removed due to a significant number of missing values. Furthermore, as the scoring criteria for TOEFL and GRE total scores may vary across different years, the scores have been standardized or converted.

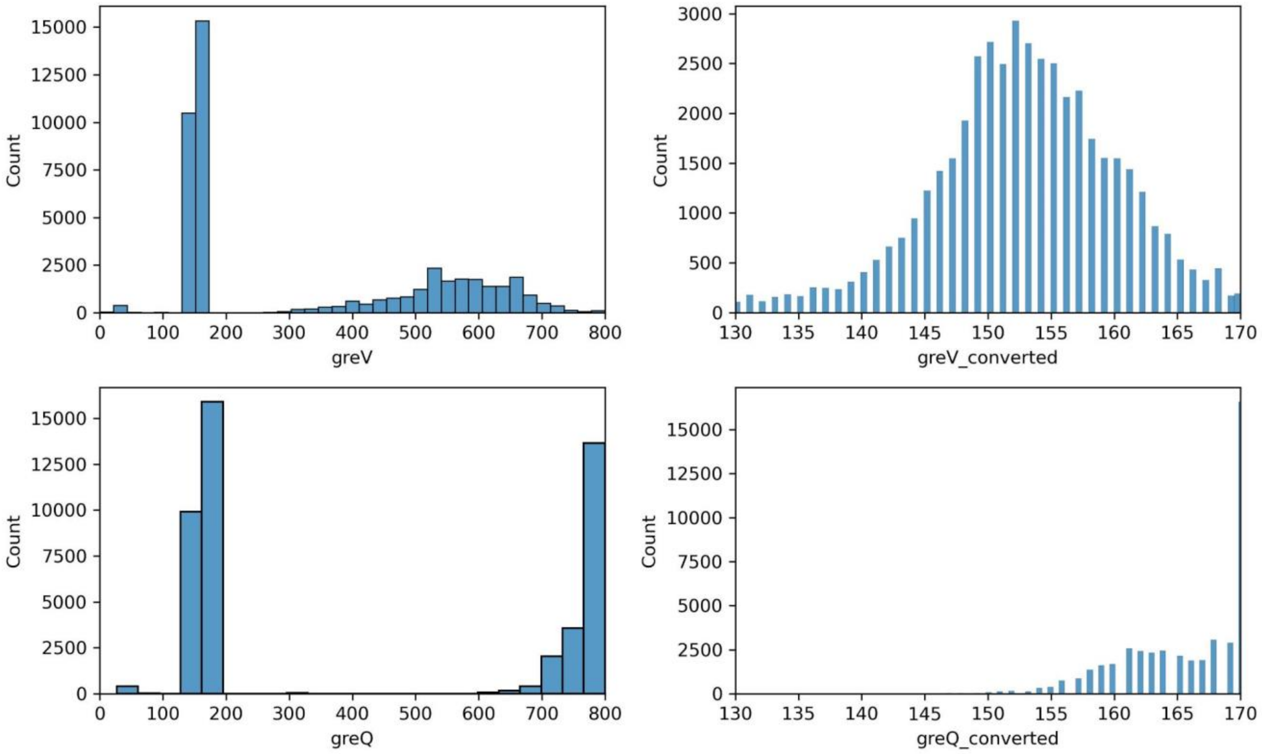


Figure 3: GRE distribution

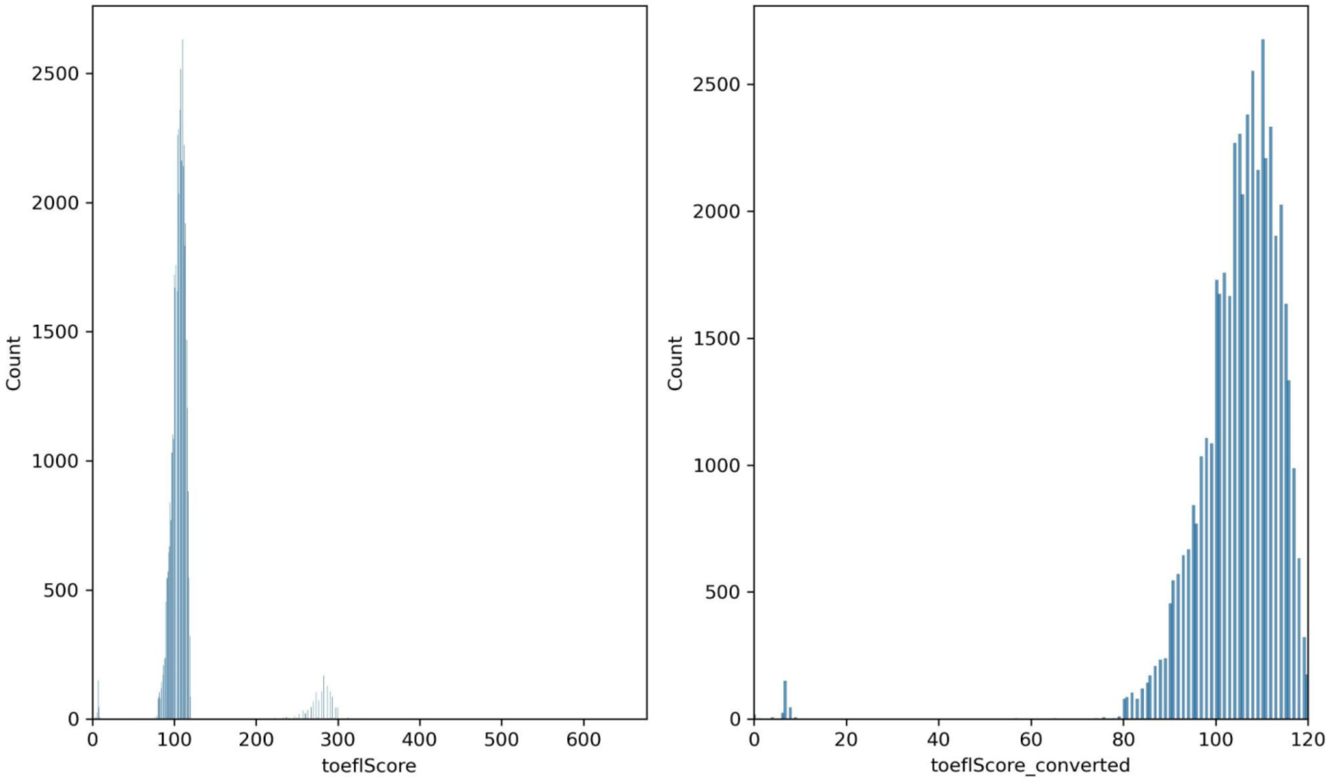


Figure 4: TOEFL distribution

As there is an abundance of attribute values within the major field, a direct encoding operation would result in an excessively sparse matrix. Therefore, a preliminary major mapping process is conducted. The majors are categorized into several broad categories, including CS (computer-related), EE (electronics-related), BA (business-related), OE (other engineering majors), SD (science), and SS (social science). A major mapping table is constructed to map all majors to these broader categories. The distribution of the number of records in each major category after the conversion is depicted in the figure 5.

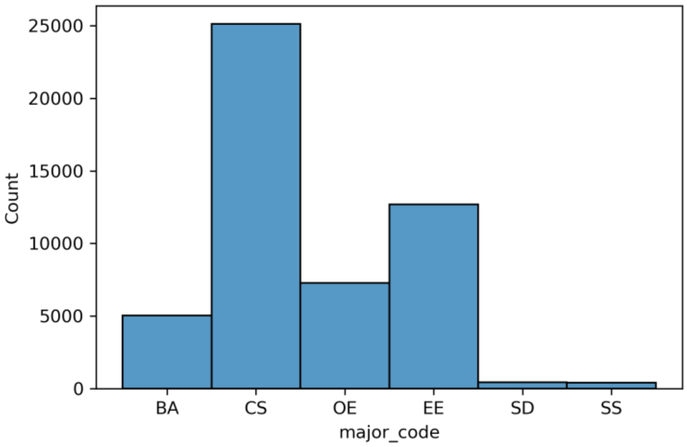


Figure 5: major code distribution

Given the extensive number of undergraduate colleges, and the absence of a standardized grading system for these institutions, undergraduate universities have been categorized into five levels based on the admission rate of undergraduate university applicants to postgraduate universities. These levels are denoted as 1, 2, 3, 4, and 5. The distribution of these levels is illustrated in the figure 9.

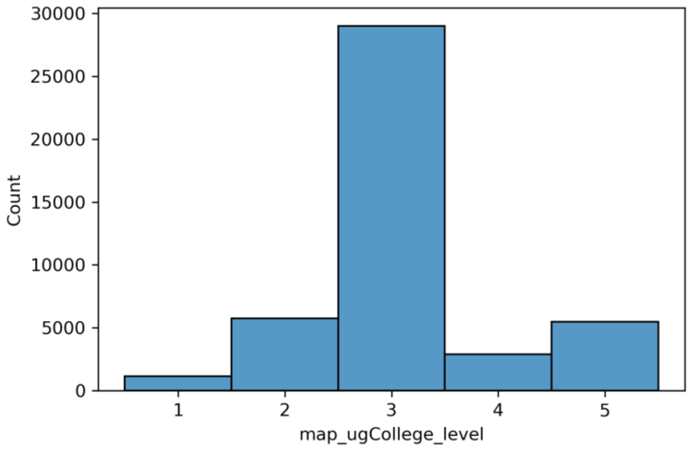


Figure 6: university distribution

3.3 KNOWLEDGE REFINEMENT

Following the acquisition of specific knowledge, a process of knowledge refinement is undertaken. The diagram illustrates that knowledge refinement is an iterative procedure. Initially, a foundational model is established, and experts assess its suitability. If there is room for improvement, newly acquired knowledge is applied to enhance the mode.

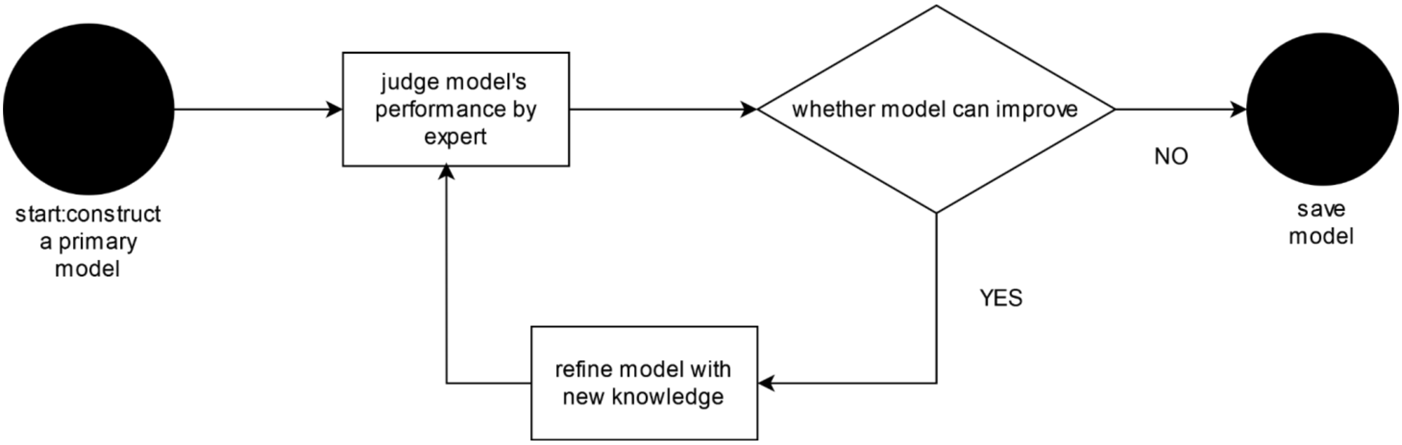

start:construct a primary model
judge model's performance by expert
whether model can improve
NO
save model
YES
refine model with new knowledge

Figure 7: knowledge refine flow chart

4. SOLUTION

The knowledge model in Section 3 allows us to develop an intelligent system based on machine reasoning to address user pain points. This knowledge base is the core of our intelligent system, providing the foundation for the chatbot to answer user queries, and enabling us to model the act of applying to universities as a predictive task. Our intelligent system offers three core functionalities as a solution for users:

1. We have developed a chatbot that can accurately answer user questions based on the latest information;

2. We have trained a model that predicts application success rates based on user background information;

3. We provide a wealth of information about universities and majors on our website for users to browse, along with a personality test to recommend suitable majors for users.

4.1 SYSTEM ARCHITECTURE

Figure8 shows the system architecture of EZO.

1. **Frontend:** We utilize React to construct the user interface, providing an interactive and dynamic user experience.

2. **Backend:** Next.js serves as the backend framework, responsible for managing API requests, ensuring efficient data transmission, and processing.

3. **Database:** MySQL is employed to manage the backend database, offering a stable and reliable data storage and retrieval service, ensuring data persistence and consistency.

4. **Web Server:** Nginx is employed as the web server, responsible for receiving and responding to users' HTTP requests. It also possesses capabilities for load balancing and managing static resources, thereby ensuring the stability and performance of the website.

5. **Data Collection:** We employ a Python scraper to gather data from the internet, enabling us to enrich our knowledge base.

6. **Intelligent Services:** We leverage Dialogflow, GPT API, and other machine learning frameworks to provide the intelligent services necessary for our application.

7. **Deployment:** The project is deployed using Docker, ensuring a standardized and scalable deployment process.

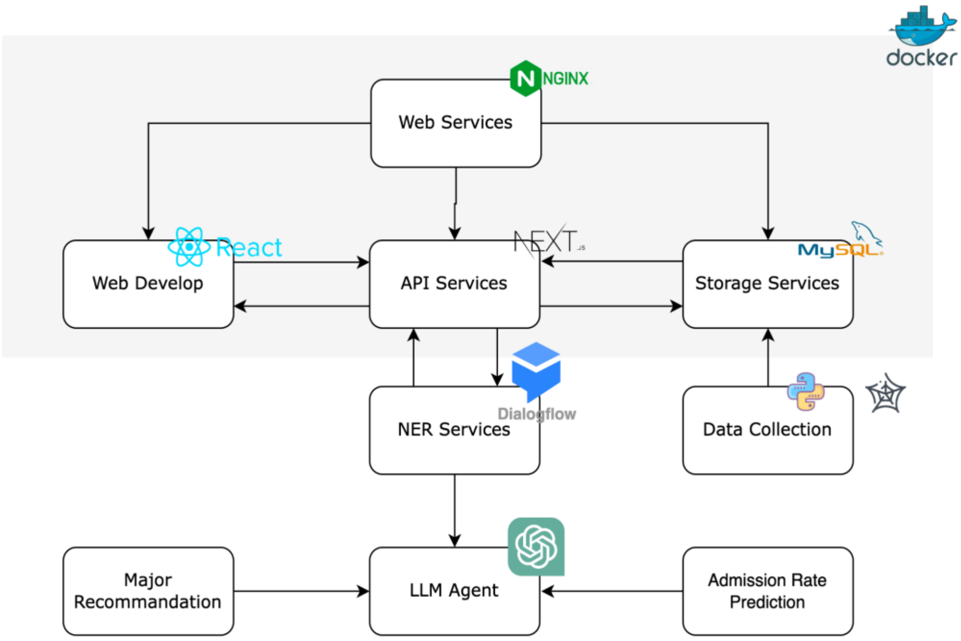


Figure 8: System Design

4.2 PROJECT SCOPE

While data mining can be performed continuously, in the context of this project, its scope is limited by the

(i) related data for universities up until October 1, 2023

(ii) date and time of data extraction

(iii) the limited data scale due to the limited computational resources.

For the chatbot, our institution data only covers the top 1000 universities according to the QS ranking in Australia. For admission rate predictions, our data covers the top 1000 universities in the United States according to the QS ranking. For browsing institution information, our data covers the top 1000 universities globally according to the QS ranking.

4.3 TECHNIQUES

4.3.1 NER

The user's intent was identified through the implementation of the Dialogflow API. Its advantages were rooted in the two core functions of Dialogflow, which encompassed entity extraction and intent matching. To facilitate entity extraction, three entity types, ‘question type", "majors within universities" and “university" have been defined. Furthermore, two intent types have been specified. A variety of training corpora were used for intention training. When a new question was posed by a user, entities were automatically extracted from the new questions by Dialogflow, and intent matching was carried out based on the training data of the extracted entities and predefined intentions. The flow chart is as follows:

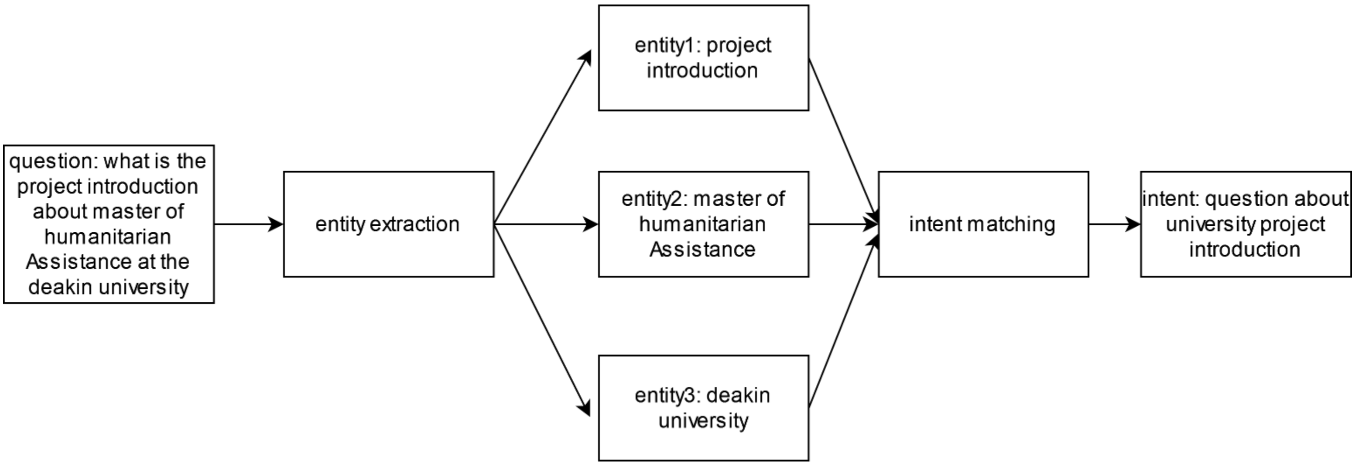

question: what is the project introduction about master of humanitarian Assistance at the deakin university
entity extraction
entity1: project introduction
entity2: master of humanitarian Assistance
entity3: deakin university
intent matching
intent: question about university project introduction

Figure 9: dialogflow flow chart

4.3.2 CHATBOT

To achieve highly personalized recommendations, we have adopted the LlamaIndex+GPT-3.5 framework. Its advantage lies in leveraging the robust data indexing and management capabilities of LlamaIndex, coupled with the advanced natural language understanding and generation abilities of GPT-3.5. This combination enables a more precise understanding and response to user needs. Not only does this framework enhance recommendation accuracy, but it also provides users with a more tailored recommendation experience, significantly elevating the system's level of intelligence.

Retrieval Augmented Generation (RAG)

LLMs are trained on enormous bodies of data but they aren't trained on your data. Retrieval-Augmented Generation (RAG) solves this problem by adding your data to the data LLMs already have access to. You will see references to RAG frequently in this documentation.

In RAG, your data is loaded and and prepared for queries or "indexed". User queries act on the index, which filters your data down to the most relevant context. This context and your query then go to the LLM along with a prompt, and the LLM provides a response.

Even if what you're building is a chatbot or an agent, you'll want to know RAG techniques for getting data into your application.

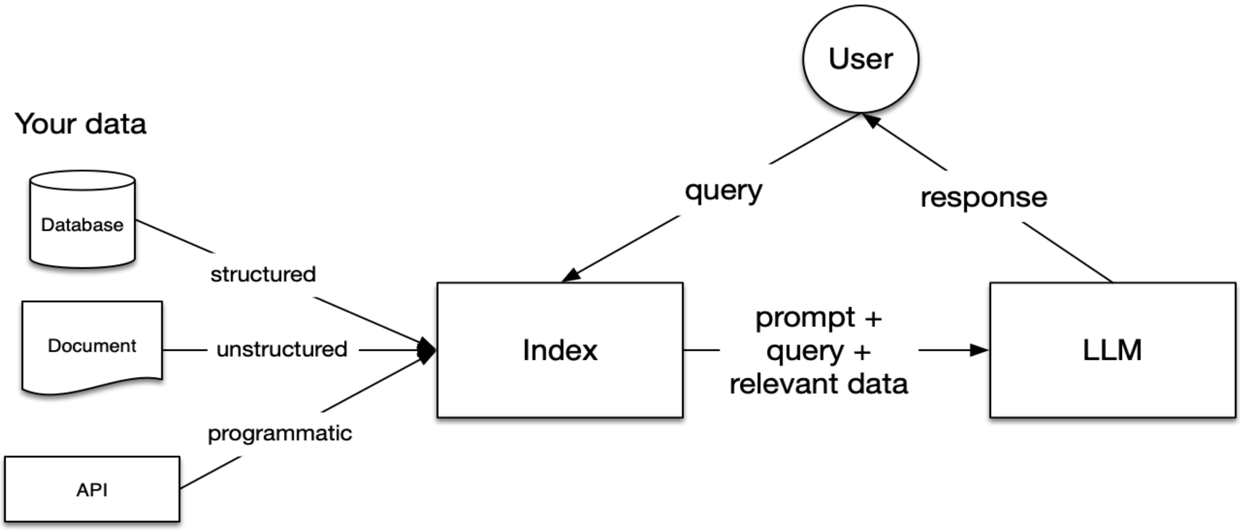


Figure 10: RAG

Stages within RAG

There are five key stages within RAG, which in turn will be a part of any larger application you build. These are:

1. **Loading**: this refers to getting your data from where it lives – whether it's text files, PDFs, another website, a database, or an API – into your pipeline. LlamaHub provides hundreds of connectors to choose from.

2. **Indexing**: this means creating a data structure that allows for querying the data. For LLMs this nearly always means creating vector embeddings, numerical representations of the meaning of your data, as well as numerous other metadata strategies to make it easy to accurately find contextually relevant data.

3. **Storing**: once your data is indexed you will almost always want to store your index, as well as other metadata, to avoid having to re-index it.

4. **Querying**: for any given indexing strategy there are many ways you can utilize LLMs and LlamaIndex data structures to query, including sub-queries, multi-step queries and hybrid strategies.

5. **Evaluation**: a critical step in any pipeline is checking how effective it is relative to other strategies, or when you make changes. Evaluation provides objective measures of how accurate, faithful and fast your responses to queries are.

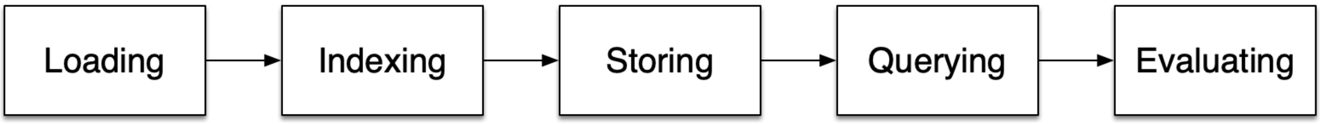


Figure 11: the flow path of RAG

4.3.3 ADMISSION RATE PREDICTION

After the data cleaning process, the cleaned data served as the input for the model. The primary advantages of using the LightGBM+Logistic Regression (LR) model for admission prediction were as follows:

1. LightGBM is capable of effectively handling missing values within the data.

2. LightGBM can seamlessly process both character features and numerical features concurrently, eliminating the necessity for one-hot encoding of character features to maintain manageable input dimensions for the model.

3. Given the online nature of this model, LightGBM offers faster prediction speeds compared to other machine learning and deep learning models.

4. LightGBM excels in capturing complex nonlinear relationships, and the output probabilities of LightGBM are integrated into the LR model. This incorporation introduces the linear component into

the Logistic Regression regression model, effectively refining the model's output and enhancing prediction accuracy.

The first figure detailed the procedure for hyperparameter optimization, while the second diagram demonstrated the model's prediction process

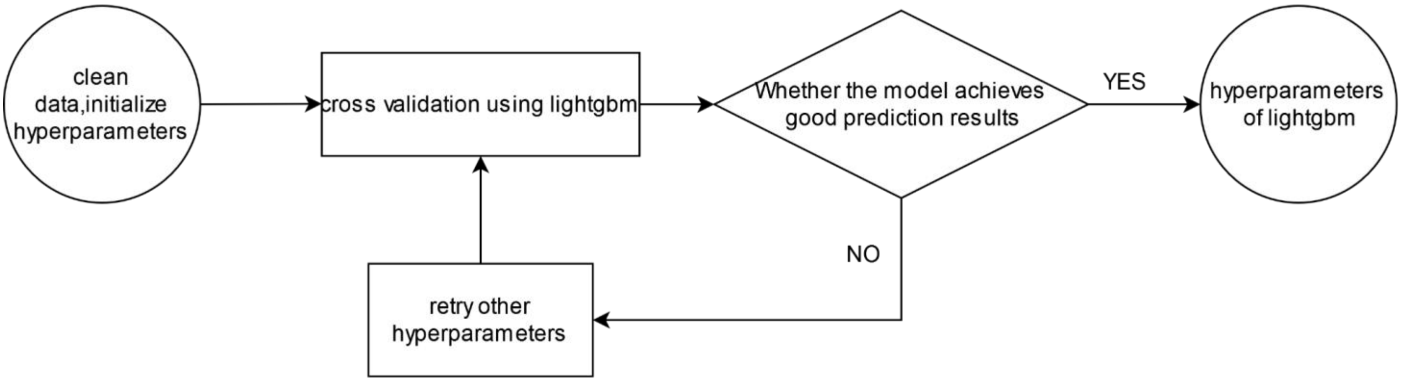

clean data,initialize hyperparameters
cross validation using lightgbm
Whether the model achieves good prediction results
YES
hyperparameters of lightgbm
NO
retry other hyperparameters

Figure 12: hyperparameter optimization

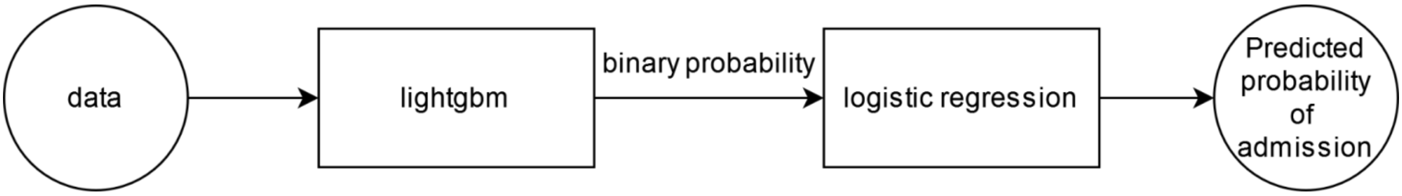


Figure 13: model prediction

A 10-fold cross-validation was performed on the LightGBM model, and the average of the prediction results was computed to determine the optimal hyperparameters for the model. Subsequently, the entire dataset was used as input for the training of the LightGBM+Logistic Regression (LR) model. The probability output by the LightGBM model was used as the input for the LR model, and the two-class label was ultimately obtained. With the hyperparameters of LightGBM set to 'max_depth': 20, 'num_leaves': 30, 'learning_rate': 0.01, 'n_estimators': 100, the model demonstrated outstanding predictive performance. Accuracy, recall, precision, and F1 values all exceed 0.7. Drawing the ROC curve, it can be observed from the curve that the AUC value was approximately 0.8, indicating that the model exhibited a strong predictive performance.

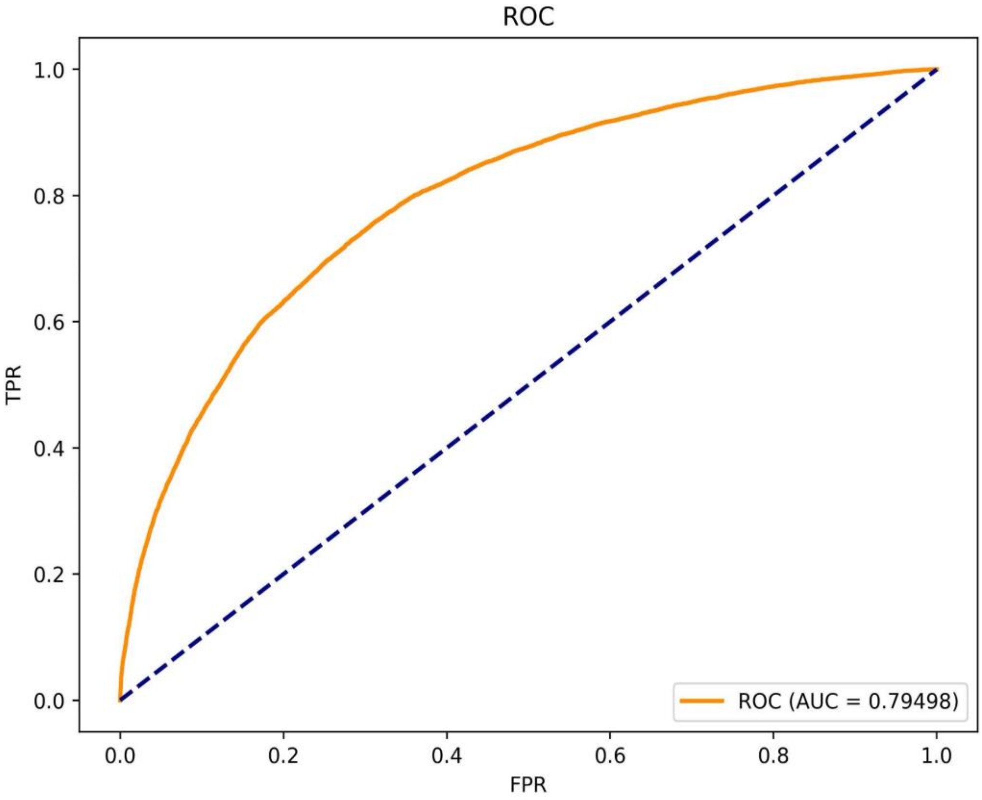


Figure 14: the flow path of RAG

4.3.4 FULL-STACK DEVELOPMENT

1. Overall Architecture

The part is primarily divided into two main modules: the "Admission Rate Evaluation" module and the "Chatbot" module. The system employs a frontend-backend separated technical architecture. The frontend is developed using React and Next.js, offering users a modern interface and interactive experience. The backend utilizes the lightweight Flask framework and is integrated with the MySQL database.

2.Technical Features

Frontend Tech Stack - React + Next.js

- **React**: A JavaScript library designed for building user interfaces. Its component-based philosophy makes the code more maintainable and expandable.
- **Next.js**: A server-rendering application framework based on React. It offers improved SEO optimization and ensures rapid initial screen rendering.

With this technical combination, the frontend can swiftly request database information, render pages, and interact with the Flask backend service.

Backend Tech Stack - Flask + MySQL

- **Flask**: A lightweight web application framework. Its simplicity ensures efficient development and maintenance.

- **MySQL**: A popular relational database management system. In this system, the Flask backend interacts with the MySQL database, such as requesting model data, storing, or retrieving information.

3.System Workflow and Interaction

Users interact with the frontend interface, where React components respond to these actions and communicate with the Next.js server.The Next.js server, upon necessary data processing, sends asynchronous RESTful API requests to the backend Flask service via the Fetch API.Upon receiving requests, the Flask backend interacts with the MySQL database to fetch or store data.The backend interacts with the respective models or requests the OpenAI API to obtain predictive results.After the backend processes the requests, it returns the data to the frontend, which then renders and presents it to the user.

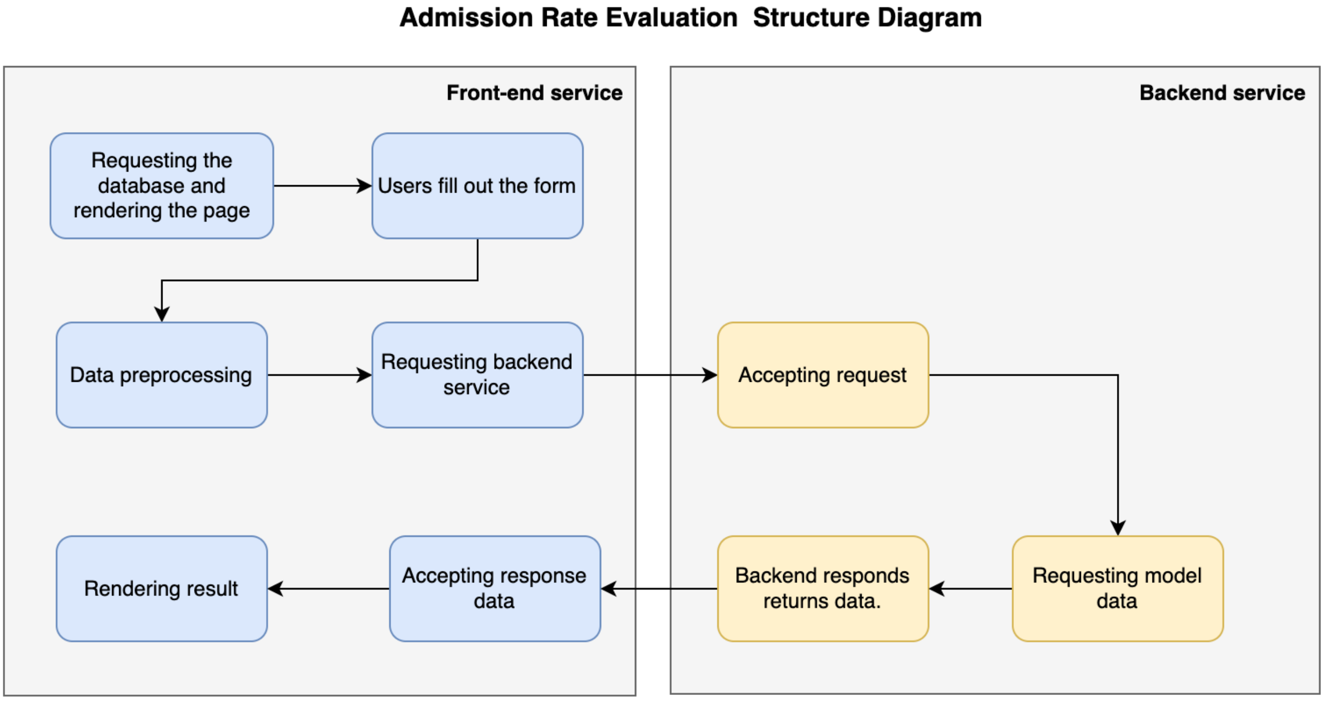


Figure 15: Admission Rate Evaluation Structure Diagram

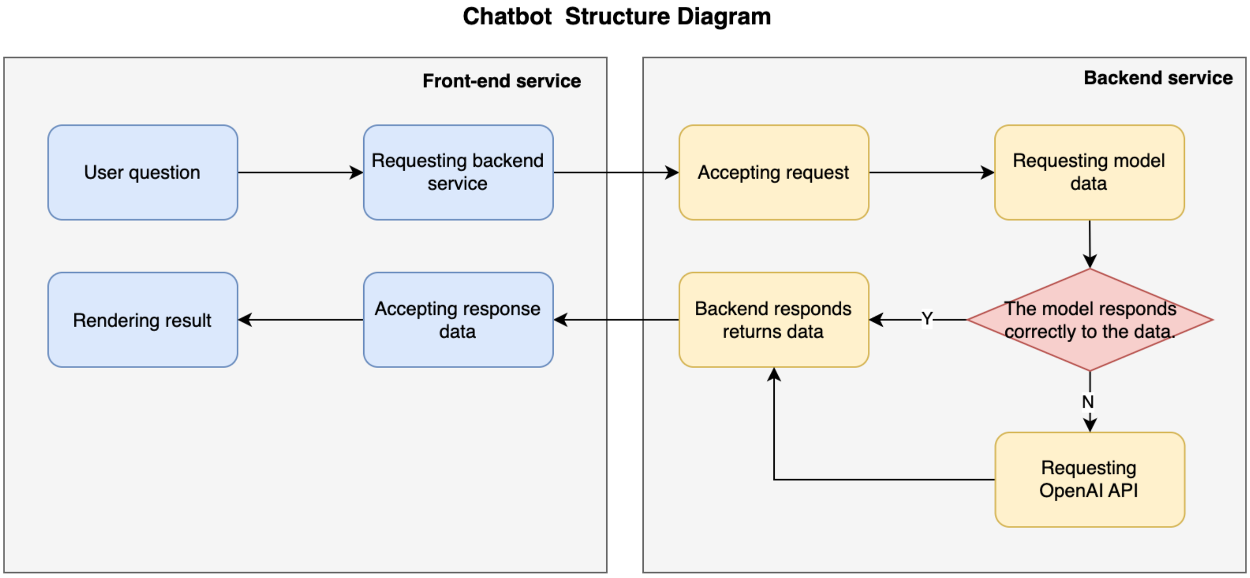


Figure 16: Chatbot Structure Diagram

4.4 ASSUMPTIONS

4.4.1 ELIGIBILITY OF USERS

While the primary audience for this project is high school students or undergraduate graduates planning to apply to overseas universities (excluding mainland Chinese institutions), we welcome users from diverse backgrounds to utilize the system. This is because it can effectively assist them in gaining insights into their individual needs and identifying universities and majors that best align with their aspirations. Users who do not fit the primary target audience can still benefit from the application success rate prediction feature by providing some of their desired background information, which will yield a rough estimate for their reference.

4.4.2 KNOWLEDGE ACQUISITION

Two types of knowledge acquisition processes have been described in section 3. They are (i) Web scrapping to obtain publicly available/documented information from websites (ii) download related admission data from kaggle website. The assumption is made that the data sourced from the website is accurate, and the acquired tacit knowledge is representative of all undergraduate students who apply to postgraduate programs. This assumption is grounded in the authenticity and authority of the data available on the website. All application data is real and is derived from a substantial sample size.

4.5 SYSTEM’S FEATURES

Despite the limitations in scope and assumptions, the EZO team, after careful consideration, has successfully implemented the most crucial functionalities in the chatbot and prediction system, providing substantial assistance to potential users.

4.5.1 SYSTEM’S INTELLIGENCE & ROBUSTNESS

Our system boasts several key features that significantly enhance its intelligence and robustness:

1. **Intent Recognition for Precise and Timely Responses**: The chatbot proficiently identifies user intents, seamlessly integrating local, accurate, and time-sensitive knowledge base to provide optimal responses. This ensures a more effective fulfillment of user needs.

2. **Contextual Memory through Vector Sequence Serialization:** Leveraging advanced vector sequence serialization technology, the chatbot has been equipped with contextual memory capabilities. This enables the chatbot to remember user information and preferences based on their historical interactions, thereby offering a more personalized and tailored user experience.

3. **Powerful Admission Success Rate Prediction Model**: Our success rate prediction model exhibits exceptional robustness. It adeptly handles a wide array of standardized test scores, including IELTS and TOEFL, as well as various grading systems for GRE and GPA. This versatility ensures accurate admission rate assessments for users, regardless of the evaluation criteria used.

These distinctive features collectively contribute to the system's intelligence and robustness, empowering it to provide tailored, accurate, and timely assistance to users throughout the university application process.

4.5.2 EASE OF ACCESS

Our intelligent system operates as a web application, ensuring accessibility to a wide user base. Any individual with internet connectivity can access the system using a personal computer, smartphone, or tablet.

Furthermore, the interface design prioritizes an optimal browsing experience for users. For instance, we have implemented numerous dynamically displayed windows and an extensive array of navigation links. Additionally, we provide input assistance in the form of multiple-choice questions, facilitating user convenience and efficiency in utilizing our system. This thoughtful design ensures that users can seamlessly navigate and engage with the platform.

4.6 LIMITATIONS

The application process for universities is highly intricate and time-consuming, with a multitude of factors that users may take into consideration, many of which ultimately influence admission outcomes. We have made extensive efforts to encompass the information that users are likely to be interested in, as well as the background details most crucial for admission results. However, it is inevitable that the system may not fully address certain users' specific needs and related backgrounds. Further enhancements to the system could involve incorporating additional information and requirements, in order to offer more comprehensive advisory insights and achieve more precise predictions.

5. CONCLUSION & REFRENCES

5.1 CONCLUSION

Through the development of EZOverseas, our team gained first-hand experience in leveraging AI to solve real-world problems. Gathering data on universities, majors and student backgrounds was crucial for training our models. We overcame challenges like missing data and highly correlated features through extensive data cleaning and feature engineering.

Our chatbot module presented unique technical hurdles. Identifying user intent from open-ended questions required carefully tuned NLP techniques. Generating concise yet informative responses while adhering to length limits tested our ability to optimize machine learning pipelines. We are proud of the personalized and up-to-date advice our system can now provide.

The university recommendation model was also very rewarding to work on. We experimented with collaborative filtering, boosting and regression to balance prediction accuracy with model interpretability. There is still room to improve by incorporating more student data. Overall, this project has equipped us with valuable skills in deploying AI responsibly to expand access to educational opportunities.

5.2 REFRENCES

[1] Koren Y, Rendle S, Bell R. Advances in collaborative filtering[J]. Recommender systems handbook, 2021: 91-142.
[2] Sharma A, Singh B. AE-LGBM: Sequence-based novel approach to detect interacting protein pairs via ensemble of autoencoder and LightGBM[J]. Computers in Biology and Medicine, 2020, 125: 103964.
[3] Li Y, Xu J, Yang M. Collaborative filtering recommendation algorithm based on KNN and Xgboost hybrid[C]//Journal of Physics: Conference Series. IOP Publishing, 2021, 1748(3): 032041.
[4] Zhang D, Gong Y. The comparison of LightGBM and XGBoost coupling factor analysis and prediagnosis of acute liver failure[J]. IEEE Access, 2020, 8: 220990-221003.
[5] Krauck A, Penz D, Schedl M. Team JKU-AIWarriors in the ACM Recommender Systems Challenge 2021: Lightweight XGBoost Recommendation Approach Leveraging User Features[M]//Proceedings of the Recommender Systems Challenge 2021. 2021: 39-43.
[6] Tang C, Luktarhan N, Zhao Y. An efficient intrusion detection method based on LightGBM and autoencoder[J]. Symmetry, 2020, 12(9): 1458.
[7] Ghosal D, Majumder N, Mehrish A, et al. Text-to-Audio Generation using Instruction-Tuned LLM and Latent Diffusion Model[J]. arXiv preprint arXiv:2304.13731, 2023.
[8] Hu Z, Lan Y, Wang L, et al. LLM-Adapters: An Adapter Family for Parameter-Efficient Fine-Tuning of Large Language Models[J]. arXiv preprint arXiv:2304.01933, 2023.
[9] Jungherr A. Using ChatGPT and Other Large Language Model (LLM) Applications for Academic Paper Assignments[J]. 2023.

6. APPENDIX

6.1 INSTALLATION AND USER GUIDE

RECOMMENDED BROWSERS

- Google Chrome
- Mozilla Firefox
- Microsoft Edge

In fact, our application is deployed using Docker, allowing it to run in any modern browser environment.

SYSTEM OVERVIEW

- Frontend built with React for responsive UI
- Backend built with Python Flask for flexibility
- MySQL database for data persistence
- Dialogflow for natural language chatbot
- Docker for standardized deployment

USER INTERFACE

- Homepage with project overview and menu
- Chatbot page for conversing with the AI assistant
- Admission odds page showing predicted chances
- University browse page to search schools and programs
- Account page to store user information

DEPLOYMENT

- Docker provides OS-level virtualization for streamlined deployment
- Docker Compose coordinates multi-container applications
- Containers created for frontend, backend, database services
- Automated build process for quick redeployment
- Horizontal scaling supported for high availability

DATA COLLECTION

- Web scraping scripts in Python gather university data
- Public datasets provide supplemental user data
- Secure protocols ensure integrity of collected data

For instructions on how to run the project, we recommend you refer to the readme file in the main directory. We will not repeat it here.

6.2 INDIVIDUAL PROJECT REPORTS

Your Name:	LI YIYAO
Contribution:	1. Led the development team in requirements analysis, feature design, and system architecture design. 2. Designed and implemented relevant algorithms and functionalities for the chatbot. 3. Completed the web scraping and pre-processing of data used for the chatbot and NER. 4. Developed some front-end pages and back-end APIs.

Learnt	1. Learned how to design complex intelligent systems, which includes requirements analysis, market research, and combining these preliminary preparations to determine the most market and user-driven functionalities. 2. Enhanced understanding of intelligent system concepts, gaining a better grasp of concepts like LLM, NER, and prediction models. 3. Developed coordination and leadership skills within the team, deepening understanding of agile development practices.
Application situation	1. The technologies I have learned in this project, including LLM fine-tuning, NER, Vector Indexing, are among the most cutting-edge and practical artificial intelligence techniques. These skills will greatly assist me in addressing complex problems in future work scenarios. 2. Taking on the role of a leader in this project has effectively honed my leadership and coordination abilities. As projects become increasingly intricate, involving larger teams, the demonstrated coordination and communication skills in this project will enable me to collaborate more effectively with teams in future work endeavors.

Your Name:	LIANG ZHU
Contribution:	In this project, I was responsible for the technical framework and coding of the "Admission Rate Evaluation" and "Chatbot" systems. The main purpose of these two systems is to provide users with an interactive interface that allows them to obtain the information and services they need. Each system contains components for both the front-end and back-end, and through the structure that I designed, ensures the smooth flow of data and effective processing of requests. 1. Technical framework design: I designed a complete technical framework for both systems, identifying the communication protocols between the front-end and back-end, as well as how the data would be processed and stored. 2. Front-end development: Using React and Next.js, I built high-performance, responsive user interfaces for both systems. Through this interface, users can easily enter information, send requests, and view results. 3. Back-end development: I handle front-end requests, and interacted with the model data. In addition, I also ensure that the backend is able to effectively communicate with recommendation systems and DialogFlow, to obtain necessary data and feedback.
Learnt	1. Technical Depth: Through this project, I have further deepened my understanding of React, Next.js and Flask. I have learned how to combine these technologies to build efficient, stable and fully functional systems. 2. System Design and Implementation: I have learned how to design and implement a complete technology system from scratch. This includes identifying requirements, selecting appropriate technologies, writing code and conducting testing. 3. AI Models and Recommendation Systems: Through

	interactions with OpenAI APIs, as well as other recommendation systems and dialogflow, I have learned how to apply AI technology to real-world projects. I have learned how to provide personalized recommendations and responses based on user needs and behavior.
Application situation	1. Technology application: The technical knowledge and practical experience I have gained will provide valuable guidance for me in other projects and work. Whether it is front-end, back-end, or overall system design, I am fully prepared. 2. Team collaboration and leadership skills: While I have taken on the main technical tasks in this project, communication and collaboration with the team are still very important. I have learned how to effectively communicate with team members and ensure the smooth progress of the project.

Your Name:	Shen Kaiyuan
Contribution:	1.Using pandas to do the data processing. After getting data of the background of students who applied for overseas graduate programs from Kaggle, I firstly dimensionality reduced the dataset and deleted those columns which contains too many missing information. Considering there are so many different majors in the dataset, I mapped the majors to broader categories (For example, Computer Science and Computer Engineering are both mapped to CS). 2.Completing the feature engineering. Undergraduate colleges are graded according to the admission rate of famous universities to solve the problem of sparse column data. Additionally, due to the change of total score, I mapped old TOEFL and GRE score to the new one proportionally. As the algorithm we choose is LighGBM, which can seamlessly process both character features and numerical features concurrently, it’s unnecessary to use methods like one-hot encoding to fit the model selected. 3.Working as my group member **Zhang Qiyuan**’s assistant, who was responsible for the training part. He is proficient at ML and really taught me a lot.
Learnt	I really benefit a lot from the project. As a transcoding student, I scarcely had any experience of programming before, for I mainly leant hardware development during my undergraduate period. Thanks to my teammates, they helped me a lot and recommended many useful online courses for me to have a better command of programming knowledge. Take machine learning as an example, my knowledge was still at the theoretical level. After a month of hard work, I can train a model successfully by myself.
Application situation	In the future when I get a job or a chance for internship, I will apply the knowledge of ML into training various models.

Your Name:	Zhu Xuanya
Contribution:	Admission rate prediction model training Web Flask backends

	Video/PPT/Presentation
Learnt	Learned the LightGBM model and learned the development interface and how to connect the front and back ends.
Application situation	It provides a foundation for front-end and back-end interface docking in future project development. Standard interface writing and related usage documentation are very important for future study and work.

Your Name:	Zhang Qiyuan
Contribution:	- Cleaned the data captured by web crawlers, designed database tables, and wrote SQL statements to store the structured data in the database tables. - Defined the intent in Dialogflow, established the school university and project information in the database as entities, generated a corpus to train each intent, enabling NER to extract entities more accurately, and matched the correct intent based on the extracted entities - Implemented the back-end interface in the Flask framework, established the SQL template, extracted distinct data from the database based on different intents, and forwarded the data to the ChatGPT interface for questioning - Established the LIghtgbm+LR model for application prediction, adjusted the optimal hyperparameters of the model, and evaluated the performance of the model
Learnt	I learnt how to choose appropriate machine learning models based on different data sets and application scenarios, How to perform hyperparameter optimization and model evaluation for the selected model.
Application situation	In future study and work, I will be able to choose appropriate machine learning models according to different application scenarios or use different models for model combination to meet actual needs.

6.3 MAPPED SYSTEM FUNCTIONALITIES

System Functionalities	Module	Knowledge/ Techniques/ Skills
System Design	Machine Reasoning	Data Driven System
Knowledge Base	Machine Reasoning	Knowledge Represenation/ Acquisition
Chatbot(Retrieval-Augmented Generation)	Reasoning Systems	Uninformed Search Techniques(Graph)
NER	Cognitive Systems	Cognitive Knowledge Representation and Reasoning