Automated Admission System Based on UCLA Graduate Dataset

Yu Liu, Shengyao Shao, Jing Song
Boston University, EC601 course project

Purpose

Our project is designed to help those graduate school admission officers to accelerate their admission process. Our web application will provide users with standardized summaries of applicant's transcript, sentiment analysis of the applicant's reference letter, and the predicted chance of admission.

Introduction

Most Viable Product:

The product should provide users with a report that summarizes the useful information presented in each applicant's transcript.

User Stories:

- I, as a graduate school admission officer, should be able to see a report of sentimental analysis on the reference letters.
- I, as a graduate school admission officer, should be able to see the applicants' chance of admission based on their application materials.
- I, as a graduate school admission officer, should be able to log into the system.

Technology Selection:

- JavaScript
- Python
- PDF Converter API
- Google Natural Language API
- Mysql

Product Components:

- Transcript Processing
- Recommendation Letter Sentiment Analysis
- Probability of admission
- Generate a summary of all above parameters for each student

Database

Database:

- All the files are saved locally, only the file path to them are stored in the database.
- MySQL is used for this project for it stores data in columns and rows. The first column of each row stores the name of the file and the second column stores the path to the file as the following graph demonstrates.

```
('b.txt', 'C:\\Users\\Vanquish\\Desktop\\testdata\\b.txt')
('c.txt', 'C:\\Users\\Vanquish\\Desktop\\testdata\\c.txt')
```

- urllib.request is imported to retrieve file content from the link.
- Firebase Google authentication is required before uploading and accessing any data

Machine Learning Algorithm

Prediction Model:

- One of the MVPs of our project is to analyze the probability of admission automatically.
- The training dataset used is inspired by the UCLA Graduate Dataset owned by Mohan S Acharya.
- Random forest regressor is used to predict chance of admission.

Preprocessing:

- The variances of applicants' GRE and TOEFL scores are relatively small, so a preprocessing is added before training.
- It will scale the scores by using maxminscaler method.
- After preprocessing, the average accuracy of the model is 97%.

File Processing

PDF Converter API:

- Users upload transcripts and recommendation letter in PDF formats
- CloudConvert API converts PDFs into TXT formats

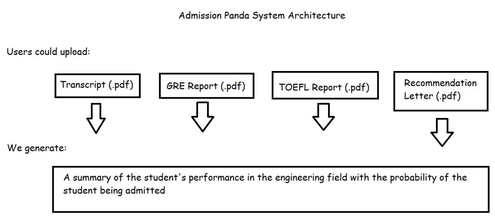
Data Scraping:

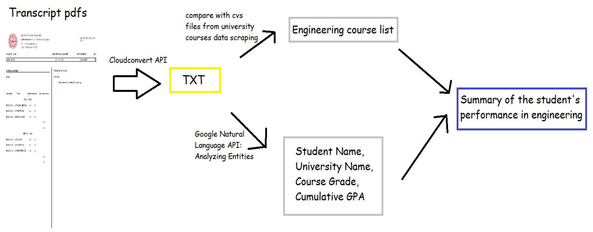
- Data of US university engineering courses scraped from university websites
- Course data stored in CSV file
- Compare courses in transcript with those in the CSV file
- Generate list of all engineering courses the student has taken in the result page

Google Natural Language Processing API:

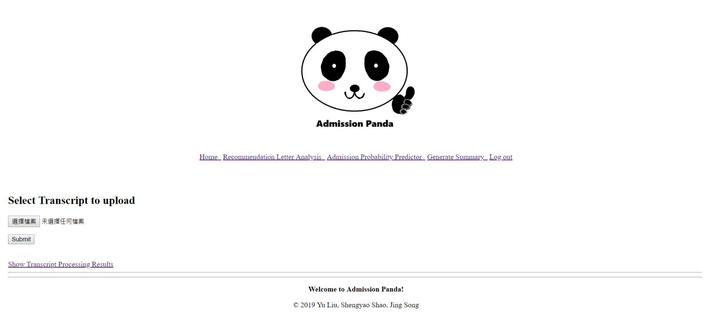
- Entity Analysis
  - Process and classify entities in the transcript
  - List entities, such as the name of the student, the name of the university, cumulative GPA, etc. in the result page
- Sentiment Analysis
  - Pass the recommendation letter in TXT format to the API for sentiment analysis
  - List the sentiment score and magnitude generated by the API in the result page

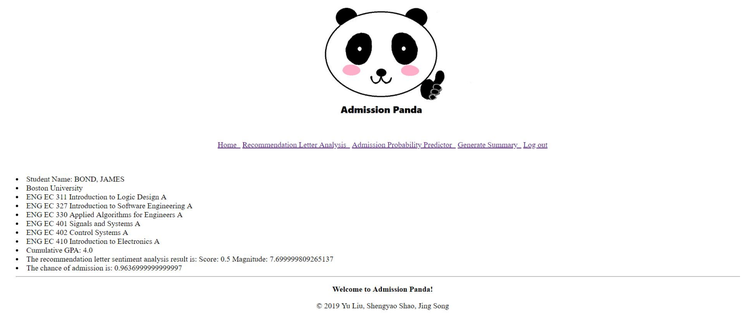
System Architecture





Result Demonstration





QR code of GitHub Repository