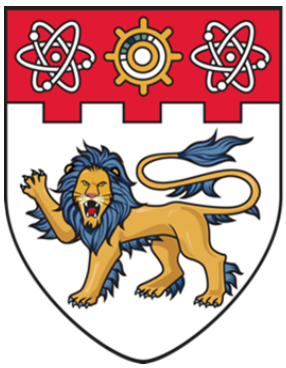


NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

CZ4125 Developing Data Products

2023-24 Semester 1

TalentTrove: Job Recommendation & Application Tracking

Group Members:

Karanam Akshit (U2020311E)

Agarwal Pratham (U2023384F)

Louis Wirja (U2023825E)

Chopra Dhruv (U2023974A)

Contribution List

All Members had an equal Contribution to this project

Karanam Akshit

- Generating Training Data for Classifier Training
- Training the 2 classifiers - Job Email Classifier and Job Stage Classifier
- Helped in linking classifiers inference code to the application tracking pipeline
- Report writing on Application Tracking Pipeline

Agarwal Pratham

- Extraction of Job Postings data from MyCareersFuture, Glassdoor Job Board, Crunchbase Company Data, Trustpilot API for Logos and Data.Gov for Singapore based company information.
- Implementation of Dashboard App backend in Python.
- Streamlit code for Dashboard.
- Combining the models in Job Tracking Pipeline and Evaluation of Models
- Implemented NER pipeline using FlanT5

Louis Wirja

- Preprocessing of MyCareersFuture job descriptions
- Implementation of Job Recommendation pipeline
- Report writing on Job Description Parsing and Cleaning, Job Recommendation and User Metrics Logging

Chopra Dhruv

- Preprocessing of Glassdoor job descriptions
- Implementation of Job Recommendation pipeline
- Report writing on Job Description Parsing and Cleaning, Job Recommendation and User Metrics Logging
- Implementation of User Metrics logging
- Competitor Analysis & Overview for Report & Video

Table of Contents

1. Introduction

In the constantly changing world of work, finding a job has become a complicated and time-consuming process for those trying to progress in their careers. Acknowledging the difficulties that come with job hunting, we introduce TalentTrove—an all-in-one job recommendation and application tracking tool that simplifies the job application process and transforms the way job seekers navigate this journey.

The main problem for job seekers is the overwhelming and often disorganized process of applying for jobs. Without a central place to manage applications, and with an abundance of information, there is a risk of missing opportunities and facing potential setbacks in achieving career goals.

TalentTrove distinguishes itself by providing a central hub for job searching, easy-to-understand interfaces, and a simple way to track applications. Our focus on user-friendly navigation, along with advanced features like integrating job descriptions from platforms like Glassdoor and MyCareersFuture Singapore to recommend suitable job postings for the user, and utilizing Gmail API for efficient application progress tracking, positions TalentTrove as an innovative solution in tackling the challenges of the job application process.

In the following parts of this report, we will explore the detailed features of TalentTrove, showcasing how each element contributes in combating the challenges associated with job searching.

2. Competitor Analysis

In this section, we will attempt to analyze some of the potential competitors for our tool and establish our differences from these existing solutions.

1) **LinkedIn (Job Search & Recommendation Competitor)**: LinkedIn is an employment centered social platform tool. It seeks to distinguish itself from the existing online job posting services by allowing users to establish Professional connections. While both LinkedIn and TalentTrove provide Job recommendations, we designed TalentTrove to primarily be an application management tool that helps streamline the process of online applications. The Recommendation tool and the Job Search page are present to help the users get started with their applications. Furthermore, we attempt to improve upon existing recommendation tools by making use of cutting edge strategies such as generation of Job Descriptions using user resumes. Zheng et. al.(2023) make use of this strategy to make recommendations more robust and explainable to the users.
2) **Jobify (Job Tracking Competitor):** Tools like Jobify help organize the application process by tracking users' applications and presenting application stages in a concise dashboard. While we agree with the need to manage applications in a streamlined

manner, tools like Jobify put the burden of organization on the users. Jobify asks the users to add the applications and deadlines manually to the tool for generating reminders. We believe in the principle of least resistance, and desire to make the application as easy to use as possible. Thus, we introduce our novel **Email-based job application tracking** feature that gathers information about the jobs that the user has applied to and the progress in these applications using our in-house Machine Learning capabilities.

3) **NUS career+ app (Local Tracking & Recommendation Competitor):** NUS career+ app is a mobile application jointly developed by NUS & Jobtech Pvt. Ltd. While we were not able to access the application ourselves (the tool is only meant for NUS students), we deduce from the feature listings that the application is mainly meant for understanding trends in the Singapore Job market. It has features for job search and recommendation as well as reminder generation for career events. While this makes the tool very personalized, we take this further by linking TalentTrove directly to the users' Email accounts. Our job tracking capabilities specific to each user make our tool stand out in comparison to the NUS career+ app.

3. Data Collection

We are collecting mainly two sets of data for our project i.e. **Job Postings and Company information dataset.**

As our platform is not meant for recruiters or companies, and is mainly focused for individuals who are looking for jobs and finding it a hassle to track their application progress while job hunting, we do not offer a solution or platform for recruiters to post jobs on the platform. We believe there are a lot of big platforms that have already offered such solutions such as LinkedIn, JobStreet, Indeed, Glassdoor, etc., so we will be maintaining our job database by regularly scraping Glassdoor. We chose to scrape Glassdoor as all of the Job Boards mentioned above tend to have similar job postings. To make it more suitable for Singapore job applications, we are also scraping daily job postings from MyCareersFuture SG.

MyCareersFutures.sg: Job postings are scraped using its publicly available API. The collection process is shown to be fast and reliable. The following steps are performed to improve performance:

Parallel Processing: Parallel requests are implemented to handle multiple API calls simultaneously. This significantly speeds up the data collection process.

Error Handling: Robust error handling is performed to deal with API rate limits and connection timeouts.

Glassdoor Job Board: Selenium along with Network Performance Capability are used to track the network logs and extract the token and cookies which are then used to authenticate the Glassdoor API. We also refreshed the tokens if they expire while scraping. For now, only STEM and Finance related jobs were scraped from Glassdoor as scraping took time due to the need of resting between concurrent calls of the API; otherwise it would have led to a network IP block. Since this process is slow and involves the avoiding of network IP blocks, we focused on optimizing the scraping process and managing network requests by performing the tasks below:

> **Rate Limiting and Timing:** We fine-tuned the delay between API calls to optimize the balance between speed and avoiding detection. Implementing a more intelligent request scheduling based on server response times and error codes can be beneficial.
>
> **Selenium Optimization:** Optimized our Selenium setup. By using headless browsing, disabling images, which improves the performance.
>
> **Expand Job Categories:** Gradually we plan to expand to more job categories beyond STEM. This can be done by incrementally including new categories and monitoring the system's stability and response.

Currently, we are using a local ChromaDB which is paired with a SQLite DB. ChromaDB is used to store vector embeddings of the job descriptions, which will be used by our Recommendation Engine and SQLite DB to store other crucial data of the Job Postings.

For the companies dataset, we collected Singapore-based companies using publicly available Data.gov.sg APIs and scraped Crunchbase Database using Simplify API to obtain the global companies dataset. For companies that we are not able to find from both databases, we have used the Trustpilot Company Search API to extract company wide data and their logos.

4. Job Description Parsing and Cleaning

4.1 Glassdoor

To utilize the recommendation system, we require that the Job Description for each role to be in a standardized format. In this project, we defined the standard Job Description format to be information about the responsibilities followed by the requirements for the role. We have purposely chosen to exclude information about company culture and diversity. Additionally, we did not include very specific information about the job posting such as Job Metadata (e.g. ID, application timelines, etc.) and contact information for queries. A sample Job Description and its Standardized form are given below:

Mined Job Description

Position Title

Artificial Intelligence AI Software Engineer – Accounting Firm

Description
3E Accounting Singapore is currently looking to appoint a Artificial Intelligence AI Software Engineer to join their company digital R&D team in Singapore office.
3E Accounting Pte Ltd aims to be the leading corporate service provider and top accounting firm in Singapore providing One-Stop Solution services for our clients.
3E Accounting Pte Ltd, Singapore is the global headquarters of 3E Accounting International, an International Accounting network with a global presence in more than 80 countries worldwide.
Our Mission
To offer the Three Es: efficiency, effectiveness and economy, all part of our One-Stop Solution services for our clients.
Our Vision
The World's leading corporate service provider, offering services beyond excellence.
Remuneration is commensurate with qualifications and work experience. Qualified and interested candidates should email their resume to info@3ecpa.com.sg
Please state your current and expected salary.
Job Responsibilities
Develop, integrate and maintain in-house Artificial intelligence (AI) using leading AI technologiesDevelop, evaluate, deploy and support automatic workflowsHands-on installation of technologies, setting up development environment, coding from scratch, automating build and deploymentWork with machine learning \/deep learning \/NLP to develop systems to the analysis of text datasets.Ensure the technical documentation and learning material is updatedTo schedule and monitor all projects in chargeAny other ad-hoc duties as required or assigned
Requirements
Degree in Computer Science, Information Technology or related degreesPrior experience in AI software developmentGood knowledge of Artificial Intelligences (AI) and Machine Learning concepts.
Hiring Organisation
Employment Type
Full Time
Industry
Accounting Firm
Job Location

51 Goldhill Plaza #07-10\/11, 308900, Novena, Central, Singapore
Working Hours

Mon – Fri: 9:00 to 18:00
Base Salary

S$ 4500 – S$ 8000
Date Posted

August 30, 2023
Valid Through

April 30, 2024
Apply now

Standardized Job Description

Responsibilities:
Develop, integrate and maintain in-house Artificial intelligence (AI) using leading AI technologies
Develop, evaluate, deploy and support automatic workflows
Hands-on installation of technologies, setting up development environment, coding from scratch, automating build and deployment
Work with machine learning /deep learning /NLP to develop systems for the analysis of text datasets.
Ensure the technical documentation and learning material is updated
To schedule and monitor all projects in charge
Any other ad-hoc duties as required or assigned

Requirements:
Degree in Computer Science, Information Technology or related degrees
Prior experience in AI software development
Good knowledge of Artificial Intelligences (AI) and Machine Learning concepts.

To obtain results in the standardized format above, we utilized ChatGPT with prompt tuning. The example prompts used for making the API calls are provided in the Appendix. We preferred ChatGPT for this Information Extraction task due its well established abilities as a Few-Shot Learner. Alternatives to this approach included Rule-based Mining and Supervised Classification. The former approach was unable to achieve good results because of the diversity of formats in the Job descriptions posted on different career pages, and the latter was impractical due to the unavailability of large scale datasets containing (JD, cleaned JD) pairs.

4.2 MyCareersFuture

For most of MyCareersFuture's job descriptions, they contain the two components in our standardized format, that is, the job responsibilities and the job requirements. Examples of the job description HTML texts can be seen in the Appendix.

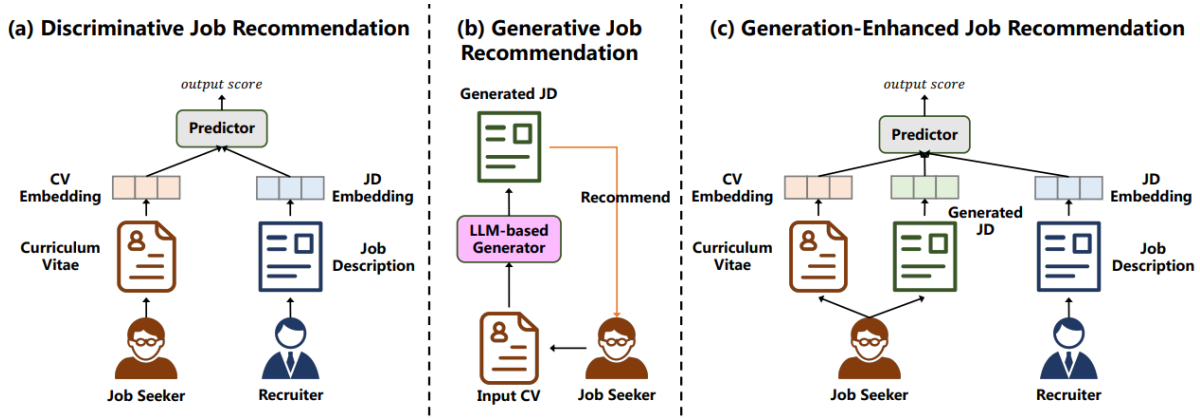
The keywords used to refer to job responsibilities and requirements are mostly consistent throughout the different job descriptions, so we have opted for a simple pattern matching logic to check whether the job descriptions contain the components we need. For retrieving the job

responsibilities, if the component contains a word from the list of keywords that represents “job requirements”, e.g. “job descriptions”, “job duties”, “job details”, etc., we then retrieve the list of items under the adjacent <ul> tag. The same is done to retrieve the job requirements, using keywords such as “qualifications” and “requirements”. Implementation of this logic can be found in the Appendix.

The items retrieved for the job responsibilities and requirements from the parsed HTML are then recombined as text and formatted in the standardized format as shown in **Standardized Job Description** under Section 3.1 to form the standardized job description for MyCareersFuture.

5. Job Recommendation

Our implementation of the job recommendation paradigm is inspired by Paradigm (b) as depicted in the diagram below. This choice was informed by Zheng et al.'s paper [1].



There are two main steps involved in the job recommendation engine. First, the user uploads their resume into the app, which is then sent to the engine for the LLM to generate a suitable job description for the resume. Second, this generated job description is queried into the vector database, and returns the top-k similar job descriptions. This two-step process thus returns similar job descriptions from the database to the uploaded resume.

We used OpenAI’s gpt-3.5-turbo to generate the job description for the resume. In our implementation of the app, we require the user to pass in their OpenAI API key to use this feature. The code (and prompt) used to generate the job description can be referred to in the Appendix.

The generated job description is then queried into the ChromaDB vector database which contains all the job descriptions. How querying works in a vector database is that the query text is first converted into an embedding vector using a pre-trained embedding model, which is then used by comparing it with the embeddings of all documents stored in the relevant collection. Similarity search is then used to find the documents with similar embeddings to the query text’s.

Here, we used the default Sentence Transformer all-MiniLM-L6-v2 model to create the embeddings.

Tasks involved in Section 3 and 4 can be summarized in a high-level diagram illustrated in the figure below:

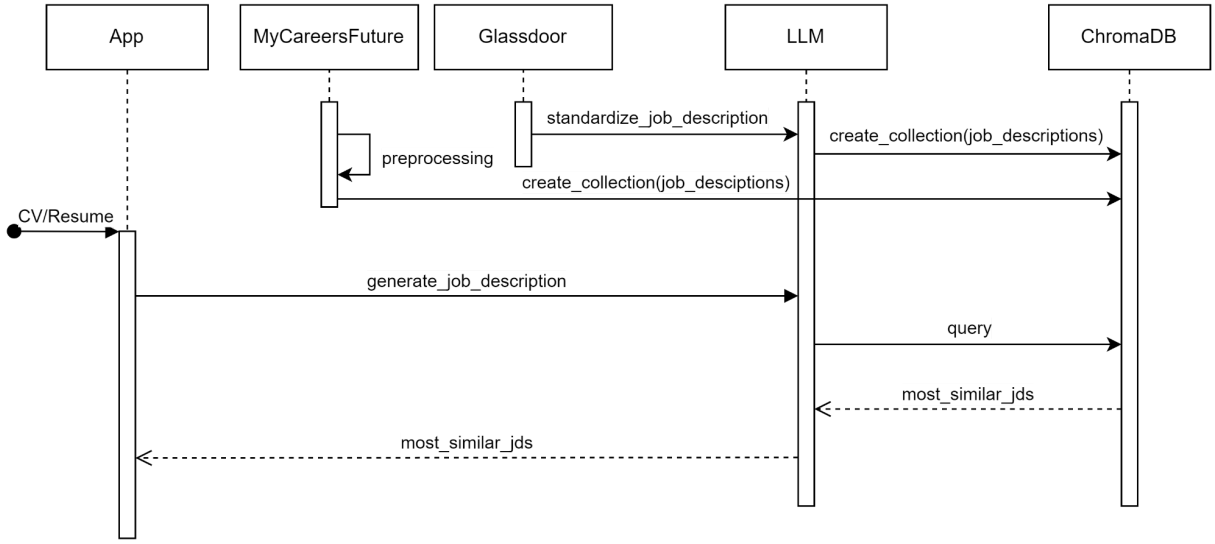


Fig. 5. Sequence diagram for preprocessing and recommending job descriptions

6. Application Tracking

6.1 Overview

The primary objective is to effectively sort crucial job-related emails sent by the employer to the applicant from the user's inbox. The system will categorize these emails into five stages: Applied, Online Assessment, Interview, Job Offer, and Rejected. These categories signify the various steps in the job application process.

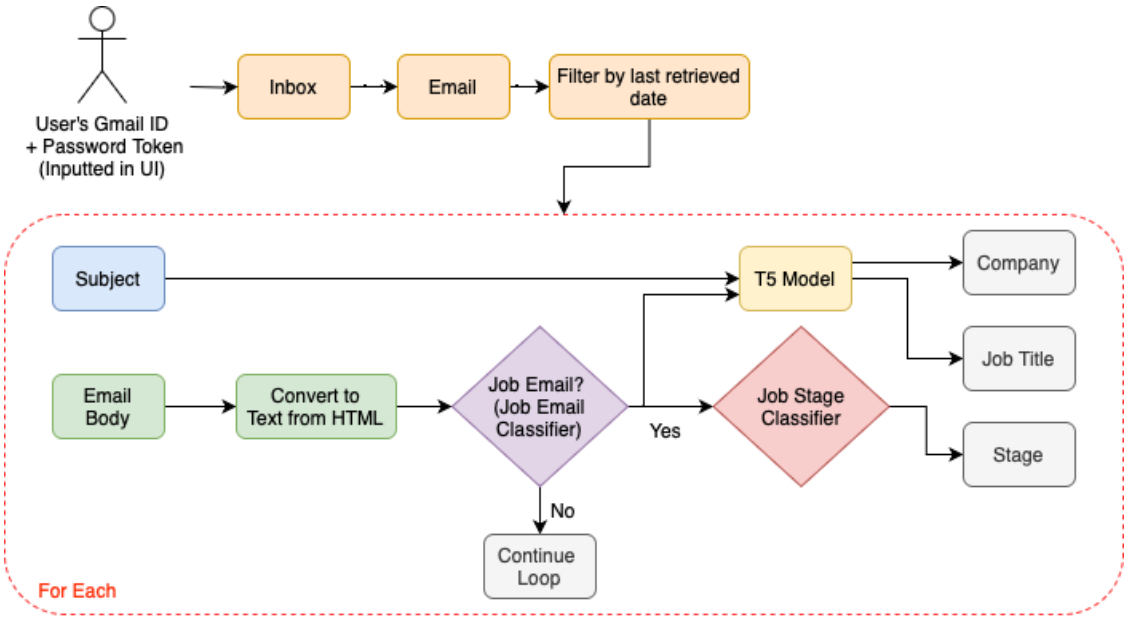


Fig. 6.1. Application Tracking Pipeline

Fig. 6.1 illustrates the pipeline outlining how relevant information is extracted from the user's email account. There are five key components in this pipeline:

Email Authentication:

The application is currently compatible only with Gmail accounts. Gmail was chosen for this project due to its widespread popularity, holding a substantial 27.6% market share (Statista, 2022). To utilize the app, users are required to input their Gmail ID and password token, obtainable by following the provided instructions, into the designated input spaces.

Inbox:

With the provided authentication, the application utilizes ‘maplib’, a Python module implementing the Internet Message Access Protocol (IMAP). IMAP stands as a standardized protocol for the retrieval and manipulation of emails on a mail server. This protocol empowers users to access their emails on the server, enabling a range of operations, including reading, deleting, moving messages, and managing folders.

In the application's workflow, all emails from the 'inbox' folder are retrieved. During the first run of the app, it processes all emails present in the inbox. Subsequent runs, however, adopt a more efficient approach by only processing emails within the date range of [last_retrieved_date, today’s_date]. This optimization minimizes redundant processing and enhances the overall efficiency of the application.

Classifying emails as 'Job Emails':

For every email that has to be processed, both the subject and body are retrieved. The email body, specifically, is utilized, and its HTML content is transformed into a textual form using BeautifulSoup. The resultant cleaned body text is subsequently input into the Job Email Classifier, where the email is classified into either 'Job Email' or 'Others' based on its content.

Assigning each 'Job Email' to its relevant stage:

For every 'Job Email' identified in the preceding step, the cleaned body text is inputted into the Job Stage Classifier. This classifier is then employed to categorize the email into one of the respective stages, namely Applied, Online Assessment, Interview, Job Offer, or Rejected.

Extracting the Company and Job Title from each identified 'Job Email':

For each 'Job Email' identified in the previous step, the subject and the cleaned body text are concatenated as follows: 'Subject: ... , Body: ...'. This concatenated text is then inputted into Flan-T5, utilizing specific prompts designed to extract the Job Title and Company Name from the email content.

6.2 Training Job Email Classifier and Job Stage Classifier

6.2.1 Set-Fit Overview

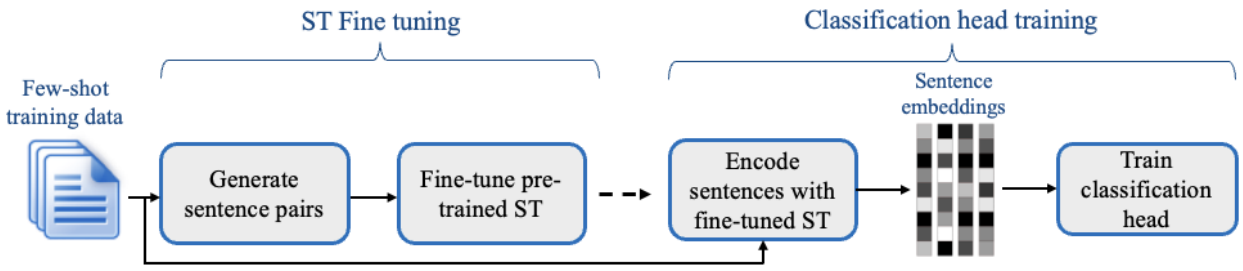


Fig. 6.2. SetFit Model Overview

SetFit (Tunstell et al., 2022) is an efficient Few-Shot Learning framework specifically designed for fine-tuning pre-trained Sentence Transformers (Reimers, Gureych, 2019). The authors of SetFit assert that, even with a small training dataset, SetFit performs competitively compared to fine-tuning RoBERTa Large on a complete training set comprising 3,000 examples.

The initial step in SetFit involves generating sentence pairs, with each pair labeled as either similar or dissimilar. Utilizing a Siamese Network, the pre-trained Sentence Transformer undergoes fine-tuning. Following the fine-tuning process, where sentence pairs are used to train

the model with a similarity loss, the resulting sentence embeddings are extracted. These embeddings are then employed to train a classification head, such as Logistic Regression.

6.2.2 Data Generation

Data was generated using ChatGPT for the training of the classifiers.

For the ‘Job Emails’ data, prompts were used to describe each job stage, and were inputted to ChatGPT. For ‘Non Job Emails’ a brief description on what topics ot genres the message should contain were prompted.

Some examples of prompts used include (some prompts are merged for easier understanding):

Job Emails:

> *‘Can you generate "Thank you for applying to... " Job email. Generate a few with different writing styles, make sure its still professional. I need atleast 10 of such examples. This is needed to train a neural classifier later. Can you output in a json format’*
>
> *‘Now can you generate in a similar format in the context of an online assessment. can you also include a few examples for cognitive tests as well. dont just use "cognitive ability", make use of other similar words as well’*

Non Job Emails:

> *‘Can you do the same for wide variety of non job email topics, (atleast 10), which include job alerts from job websites such as LinkedIn or JobsDB.’*
>
> *‘can you generate 20 more of such examples, it should vary on different topics, and should try to encompass the top topics in email writing’*
>
> *‘Also include personal emails from loved ones, company bank statements , and receipts, bills from shops’*

Making use of hard examples in prompts:

In specific instances, ChatGPT may encounter challenges in generating email samples that closely mimic real-world scenarios. While the generated content might capture the essence of the prompt given, there can be variations in the structure and language compared to real-world situations.

For instance, generating a representative sample for 'Job Alerts,' which should be classified as a Non Job Email, posed a difficulty. To address this, some examples were extracted from personal

emails, specifically those related to Job Alerts from platforms like LinkedIn. These examples were then utilized as input to ChatGPT with a prompt resembling the following:

> *'also can you generate job alerts that look like this:*
>
> *Your job alert for machine learning*
> *3 new jobs in Singapore match your preferences.*
> *Singapore University of Technology and Design (SUTD)*
> *See all jobs'*

By incorporating these hard examples, we successfully enhanced the quality of our generated dataset. All the prompts and the ChatGPT outputs can be obtained through the link provided in the appendix.

Table 6.3 shows the count on the Data Generated that was used for training the 2 classifiers using SetFit.

Job Email	Applied	20	107
	Online Assessment	30	
	Interview	20	
	Offer	20	
	Rejected	17	
No Job Email			53

Table 6.3. Generated Data Count

6.2.3 Training SetFit Classifiers

For this project, the publicly available 'setfit' package was employed, which encompasses the entire fine-tuning and classification pipeline. The 'setfit' package facilitated the implementation of both the Job Email Classifier and the Job Stage Classifier.

Table 5.4 below describes the model hyperparameters used for training both the Job Email Classifiers and Job Stage Classifier

Pre-Trained Sentence Transformer Model	paraphrase-mpnet-base-v2
Iterations (Used for generating the Sentence	30

Pairs, for Siamese Network Training)	
Number of Epochs	1
Loss Function	CosineSimilarity
Learning Rate	0.01
Batch Size	4

Table 6.5. Model Hyperparameters

6.3 Job Title and Company Name extraction using Flan-T5

6.3.1 Overview of Flan-T5

Flan-T5 represents an open-source Language Model (LLM) designed for commercial applications. Developed by Google researchers, this encoder-decoder model has undergone pre-training on diverse language tasks, using both supervised and unsupervised datasets. The primary objective of its training is to grasp the relationships between sequences of text, essentially functioning as a text-to-text model. What distinguishes Flan-T5 from other models is its training methodology centered around prompting. Essentially, the model possesses the capability to execute specific tasks like summarization, classification, and translation. For example, if you input this blog post into Flan-T5 and instruct it to "summarize this article," it comprehends the task as generating a condensed version of the content. If you acquire the model from Hugging Face, it becomes readily applicable for various general-purpose tasks.

6.3.2 Using Flan-T5

First, the subject and the body of the email are concatenated. A short question (or prompt) and the concatenated email text form the input to the Flan-T5 model. Below are the ways we used to get the Job Title and Company Name from the email text.

To get the Job Title:

```
question =  "What is the job title?If not available output None",
input_text = f"question: {question} context: {email_text}"
```

To get the Company Name:

```
question =  "What is the name of the company?If not available output None",
input_text = f"question: {question} context: {email_text}"
```

6.4 Testing and Results

The past 7 days of emails were retrieved (retrieval was done on 1st December) from one of our group member gmail account.

6.4.1 Job Email Classifier

Table 6.5 show some statistics while Figure 5.6 and 5.7 shows some examples on how the textual data looks like for the past 7 days.

	Actual	Predicted
No. of Job Emails	10	10
No of Non-Job Emails	32	32

Table 6.5. Job Email Classifier Results

	text	job
28	Subject: Nomura Application Update. Body: Dear...	1
30	Subject: ResMed Interview Invitation - Data Sc...	1
31	Subject: ResMed Offer: Data Science Intern_Lou...	1
32	Subject: IHiS Internship Offer - NLP modelling...	1
33	Subject: IHiS Internship Interview - 10 Aug, 3...	1
34	Subject: Deloitte Consulting - Intern, Artific...	1
35	Subject: Your assessment for Student Intern: A...	1
36	Subject: Your Application for Software Develop...	1
40	Subject: Thank you for applying to Squarepoint...	1
41	Subject: Your Zoom Application: Research Scien...	1

Fig. 6.6. Examples of Job Emails

	text	job
0	Subject: 🌟 Checking in from Simplify!. Body: H...	0
1	Subject: Join 39 Makers at "H2O Gen AI Day tom...	0
2	Subject: What's next for crypto? . Body: Coinb...	0
3	Subject: WLD purchase request confirmed. Body:...	0
4	Subject: Improving Retrieval Performance in RA...	0
5	Subject: Power stocks hit new highs | India ca...	0
6	Subject: This message has a gift! 🎁. Body: Ema...	0
7	Subject: Your daily dose of handpicked stories...	0
8	Subject: SINGAPORE SALVAGE ENGINEERS PTE LTD: ...	0
9	Subject: Sam Altman interview . Body: Sam Altm...	0
10	Subject: 🚀 Coinhako at a glance - Nov 2023. Bo...	0

Fig. 6.7. Examples on Non Job Emails

As indicated in Table 5.5, the classifier demonstrates a great performance in distinguishing between Job and Non-Job emails, achieving a 100% accuracy rate. It's worth noting that the precision of detecting Job emails is challenging to quantify given the small test set comprising only 10 Job emails. Nevertheless, the classifier excels in its ability to accurately identify Non-Job emails, correctly classifying all 32 emails, which is a significant test size, in this category.

6.4.2 Job Stage Classifier

As indicated in Table 6.8 and Figure 6.9, the Job Stage Classifier behaves quite well for the previously obtained 10 Job emails. There was only 1 misclassification, where the a job offer was misclassified as an online assessment stage. Upon a closer inspection on this particular email, we found out that the email also contained content where the employer was asking the job seeker to send some signed documents.

“ In order for us to set up your records timely, we will need you sign these documents along with your Employment Contract. Request you to send a copy of the following documents to my email ID”

From this example, it can be seen that since it contains words such as ‘sign these documents’ and ‘send to my email ID’, which is quite similar to what might appear in an email containing information for an Online Assessment.

Stage (Label)	Actual	Predicted
Applied (0)	2	2
Online Assessment (1)	1	1
Interview (2)	4	4
Offer (3)	2	2
Rejected (4)	1	1

Table 6.8 Job Email Classifier Results

	precision	recall	f1-score	support
0	1.00	1.00	1.00	2
1	0.50	1.00	0.67	1
2	1.00	1.00	1.00	4
3	1.00	0.50	0.67	2
4	1.00	1.00	1.00	1
accuracy			0.90	10
macro avg	0.90	0.90	0.87	10
weighted avg	0.95	0.90	0.90	10

Figure 6.9 Classification Report for the Job Stage Classifier

6.4.3 Flan-T5

Table 6.10 below shows the extraction of Company name and Job Title from the email text using Flan-T5

Actual Company Name	Extracted Company Name	Actual Job Title	Extracted Job Title
Nomura	Nomura	2024 Group Data Office Graduate Program - Singapore	Graduate Program - Singapore
ResMed	ResMed	Data Science Intern	Data Science Intern
ResMed	ResMed	Data Science Intern	Data Science Intern
IHiS	IHiS	NLP modelling & visualisation on clinical textual data	Senior Executive, Talent Acquisition Human Capital Management
IHiS	IHiS	Internship	IHiS
Deloitte Consulting	Deloitte Consulting	Intern - Artificial Intelligence & Data - Singapore	Intern - Artificial Intelligence & Data - Singapore
SAP	SAP	Student Intern: AI Engineer	AI Engineer
SAP	SAP	Software Developer Intern (SAP AI)	Software Developer Intern (SAP AI)
Squarepoint Capital	Squarepoint Capital	Desk Quant Analyst	Desk Quant Analyst
Zoom	Zoom	Research Scientist	Research Scientist

Table 6.10 Extraction of Company Name and Job Title

Flan-T5 demonstrates excellent capabilities in recognizing both the Company Name and Job Title. Nonetheless, it encountered difficulties in two instances where the Job Title was not explicitly mentioned in the email, leading to the misidentification of the recruiter's position as the Job Title.

6.5 Grouping Multiple Job Emails

Given that we are monitoring the progression of job applications, emails typically arrive in a sequential order. When the Company Name and Job Title match, the corresponding emails are grouped together. This grouping forms a chronological timeline of events, providing a structured representation of the application process. Figure 6.11 shows how the timeline is presented on the app.

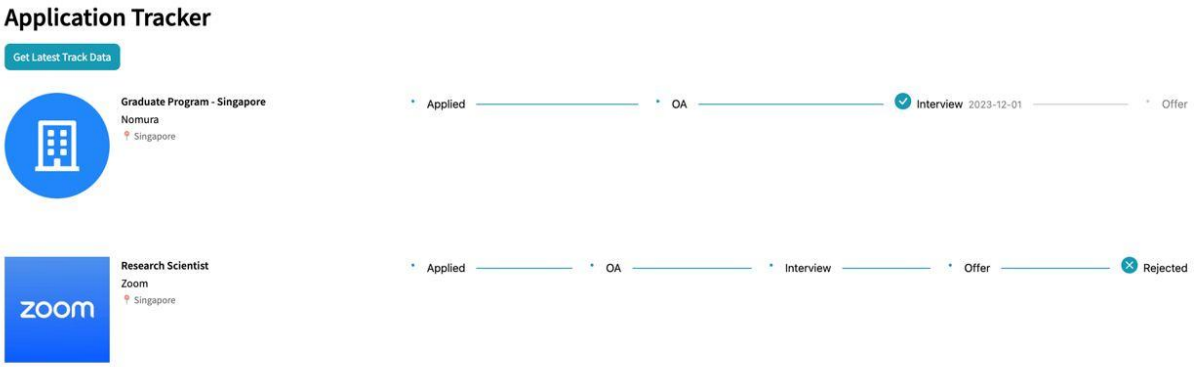


Figure 6.11 Timeline of Applications

7. User Metrics Logging

Our app also includes a logging feature, where logs are created at every event. Some of these events include: rendering of a page, entering of OpenAI API key, creation of vector database, generation of job description for uploaded resume, etc.

app.log

Reveal Now Clear Reload Share Search

```
2023-12-01 18:13:55,251 Add of existing embedding ID: 9991187
2023-12-01 19:35:07,741 TalentTrove Homepage Rendered
2023-12-01 19:35:13,153 Job Tracking Page Rendered
2023-12-01 19:39:42,647 Dashboard Rendered
2023-12-01 19:39:43,644 Matches Rendered
2023-12-01 19:40:04,327 Matches Rendered
2023-12-01 19:40:10,820 TalentTrove Homepage Rendered
2023-12-01 19:43:59,612 TalentTrove Homepage Rendered
2023-12-01 19:43:59,641 Gmail Email Entered
2023-12-01 19:45:03,809 TalentTrove Homepage Rendered
2023-12-01 19:45:03,817 Gmail key Entered
2023-12-01 19:45:03,818 Gmail Email Entered
2023-12-01 19:46:02,767 TalentTrove Homepage Rendered
2023-12-01 19:46:02,776 OpenAI API key Entered
2023-12-01 19:46:02,777 Gmail key Entered
2023-12-01 19:46:02,778 Gmail Email Entered
2023-12-01 19:46:04,822 Matches Rendered
2023-12-01 19:46:58,728 Matches Rendered
2023-12-01 19:46:58,808 Anonymized telemetry enabled. See                    https://
docs.trychroma.com/telemetry for more information.
2023-12-01 19:46:59,224 Document Read
2023-12-01 19:47:51,037 HTTP Request: POST https://api.openai.com/v1/chat/completions "HTTP/
1.1 200 OK"
2023-12-01 19:47:51,061 CV Generated
2023-12-01 19:56:30,191 Matches Rendered
2023-12-01 19:56:30,250 Document Read
2023-12-01 19:57:09,593 HTTP Request: POST https://api.openai.com/v1/chat/completions "HTTP/
1.1 200 OK"
2023-12-01 19:57:09,603 CV Generated
2023-12-01 19:57:48,133 Matches Rendered
2023-12-01 19:57:48,227 Document Read
2023-12-01 19:58:22,187 HTTP Request: POST https://api.openai.com/v1/chat/completions "HTTP/
1.1 200 OK"
2023-12-01 19:58:22,199 CV Generated
```

Logs can offer valuable insights into user interactions with the application. By recording events related to user actions, this information can be used to analyze user behavior, preferences and patterns, informing decisions for user experience enhancements.

Logging can also contribute to the A/B testing for implementing new features. Through meticulous logging, developers can conduct A/B tests by comparing two versions of a feature to determine which performs better based on user interactions. By comparing the logs from both versions, developers can quantify the impact of the new feature on user behavior.

In this way, logging not only provides insights into user behavior but also serves as a foundation for data-driven decision-making, enabling developers to iterate on features and continuously improve the application based on real user interactions and preferences.

8. Conclusion and Discussion

We have established how we have implemented our application, by first scraping job data and company information from Glassdoor and MyCareersFuture, preprocessing the data, and then using that data for the features of our application such as job recommendation and application tracking. Given an uploaded resume, the application is able to recommend some job postings suitable to the resume. The application is also able to identify which jobs the user has applied to and in what stage they are in for each job application by reading the email in the user's Gmail account.

We have also explained how our solution differs from other competitors, and why ours can have an advantage in alleviating the problems faced in the job application process.

Challenges Faced and Future Solutions

In the Application Tracking pipeline, we've achieved notable success with our model, excelling in accurately identifying Jobs and Non Job Emails. The model proficiently classifies each Job email into the correct stages. However, attaining this level of accuracy posed challenges, particularly as ChatGPT responses occasionally deviated from the ground reality. Training the classification model by concatenating the subject with the email body introduced issues, leading to misclassifications as the model disproportionately focused on the subject content.

Through extensive experimentation with various pre-trained Sentence Transformer models and employing diverse data generation strategies, we ensured the development of robust classification models. It is anticipated that there may be misclassifications, considering that the generated training data might not encompass all the diverse operating procedures employed by various companies. The use of actual data in a substantial quantity for training, as opposed to generated data, could result in a more comprehensive classification model.

9. References

1. Zheng, Z., Qiu, Z., Hu, X., Wu, L., Zhu, H., & Xiong, H. (2023). Generative job recommendations with large language model. *arXiv preprint arXiv:2307.02157*.
2. Zhu, Y., Yuan, H., Wang, S., Liu, J., Liu, W., Deng, C., ... & Wen, J. R. (2023). Large language models for information retrieval: A survey. *arXiv preprint arXiv:2308.07107*.
3. Ding, N., Hu, S., Zhao, W., Chen, Y., Liu, Z., Zheng, H. T., & Sun, M. (2021). Openprompt: An open-source framework for prompt-learning. *arXiv preprint arXiv:2111.01998*.
4. Tunstall, L., Reimers, N., Jo, U. E. S., Bates, L., Korat, D., Wasserblat, M., & Pereg, O. (2022). Efficient few-shot learning without prompts. *arXiv preprint arXiv:2209.11055*.
5. Reimers, N., & Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

10. Appendix

a. APIs used during Data Collection

- https://api.simplify.jobs/v2/company?size=100&page=1
- https://api.mycareersfuture.gov.sg/v2/search
- https://api-production.data.gov.sg/v2/public/api/collections/2/metadata
- https://glassdoor.sg/graph

b. Prompt Tuning for Information Extraction

system = '''
Clean the job description below. Do not provide any information about the company culture. Do not provide any information about diversity and inclusion. Do not provide any contact information. Output only the responsibilities and requirements. Output NOT AVAILABLE if the information is not available. Do not provide any extra information not present in the given job descriptions.
'''

user1 = '''
Machine Learning Engineer (Recommendation), BytePlus - 2024 Start
SingaporeRegularR&D - Machine learningGraduates- 2024 StartJob ID: A174093
Responsibilities
ByteDance will be prioritizing applicants who have a current right to work in Singapore, and do not require ByteDance's sponsorship of a visa. Founded in 2012, ByteDance's mission is to inspire creativity and enrich life. With a suite of more than a dozen products, including TikTok, Helo, and Resso, as well as platforms specific to the China market, including Toutiao, Douyin, and Xigua, ByteDance has made it easier and more fun for people to connect with, consume, and create content. Why Join Us Creation is the core of ByteDance's purpose. Our products are built to help imaginations thrive. This is doubly true of the teams that make our innovations possible. Together, we inspire creativity and enrich life - a mission we aim towards achieving every day. To us, every challenge, no matter how ambiguous, is an opportunity; to learn, to innovate, and to grow as one

team. Status quo? Never. Courage? Always. At ByteDance, we create together and grow together. That's how we drive impact - for ourselves, our company, and the users we serve. Join us. About the Team BytePlus Recommend helps our user to build a delightful discovery experience. BytePlus Recommend aims to provide this service to our users as a flexible and efficient tool to incorporate into their business. Through our services, users will be able to build better user experiences and grow their business further. It enhances every step of the user journey with personalized recommendations. The team provides rich personalized experiences using the latest innovations in machine learning to achieve higher retention, engagement, and revenue, explore flexible recommendation models that cater to our users' specific scenarios and goals, and explore all the need to get users started in an easy-to-use platform. We are looking for talented individuals to join us in 2024. As a graduate, you will get unparalleled opportunities for you to kickstart your career, pursue bold ideas and explore limitless growth opportunities. Co-create a future driven by your inspiration with ByteDance. Candidates can apply to a maximum of two positions and will be considered for jobs in the order you apply. The application limit is applicable to ByteDance and its affiliates' jobs globally. Applications will be reviewed on a rolling basis - we encourage you to apply early. Responsibilities - Build industry leading recommendation system for customers; - Understand machine learning techniques and business objectives, end-to-end own recommendation system and responsible for performance, improve recommendation models and strategies; - Conduct cutting-edge research in machine learning algorithms, and apply the technology to different business scenarios. - Work with cross functional teams to grow Enterprise Partnership & Solutions business. - Support the production of scalable and optimised AI/machine learning (ML) models - Focus on building algorithms for the extraction, transformation and loading of large volumes of realtime, unstructured data to deploy AI/ML solutions from theoretical data science models - Run experiments to test the performance of deployed models, and identifies and resolves bugs that arise in the process - Work in a team setting and apply knowledge in statistics, scripting and programming languages required by the firm - Work with the relevant software platforms in which the models are deployed

Qualifications

- Final year or recent graduate with a background in Software Development, Computer Science, Computer Engineering, or a related technical discipline - Experience in one or more of the following areas: machine learning, recommendation systems, data mining or other related areas; - Solid programming skills, good programming styles and working habits - Able to solve problems independently, good teamwork and communication skills; - Passion for technologies and solving challenging problems. Preferred qualifications - Experience in large-scale recommendation systems ByteDance is committed to creating an inclusive space where employees are valued for their skills, experiences, and unique perspectives. Our platform connects people from across the globe and so does our workplace. At ByteDance, our mission is to inspire creativity and enrich life. To achieve that goal, we are committed to celebrating our diverse voices and to creating an environment that reflects the many communities we reach. We are passionate about this and hope you are too. By submitting an application for this role, you accept and agree to our global applicant privacy policy, which may be accessed here: https://jobs.bytedance.com/en/legal/privacy. If you have any questions, please reach out to us at apac-earlycareers@bytedance.com

'''

assistant1 = '''

Responsibilities:

Build industry leading recommendation system for customers

Understand machine learning techniques and business objectives, end-to-end own recommendation system and responsible for performance, improve recommendation models and strategies

Conduct cutting-edge research in machine learning algorithms, and apply the technology to different business scenarios

Work with cross functional teams to grow Enterprise Partnership & Solutions business
Support the production of scalable and optimised AI/machine learning (ML) models
Focus on building algorithms for the extraction, transformation and loading of large volumes of realtime, unstructured data to deploy AI/ML solutions from theoretical data science models
Run experiments to test the performance of deployed models, and identifies and resolves bugs that arise in the process
Work in a team setting and apply knowledge in statistics, scripting and programming languages required by the firm
Work with the relevant software platforms in which the models are deployed

Requirements:
Final year or recent graduate with a background in Software Development, Computer Science, Computer Engineering, or a related technical discipline
Experience in one or more of the following areas: machine learning, recommendation systems, data mining or other related areas
Solid programming skills, good programming styles and working habits
Able to solve problems independently, good teamwork and communication skills
Passion for technologies and solving challenging problems. Preferred qualifications
Experience in large-scale recommendation systems
'''

user2 = '''
Summary
Posted: 2 Nov 2023
Role Number:200517700
The Digital apps and Diagnostics Engineering team is responsible for building the applications and services that power personal support experiences for all of Apple customers worldwide. We enable experiences in Apple Support app, Get Support, My Support. and Agreement Sales web apps, and services integrated into the Settings app on iOS and Mac including our Diagnostic Platform which is the heart of Apple's device diagnostics ecosystem connecting a billion plus customer devices. We work early with every new Apple device and OS release to ensure diagnostic readiness at product launch. We are seeking a dedicated, hard-working individual to help maintain and monitor these applications and services and drive incident management process across various multi-functional groups.

Are you a lifelong learner? Do you have a passion for troubleshooting issues and help resolve complex problems working with several cross-functional teams? Come and join us in a fun-loving, highly motivated team and be a part of Apple's support journey.

The DevOps engineer has deep domain knowledge of these products across the whole stack, creative problem solving skills, and great cross-functional collaboration skills. The candidate should be able to troubleshoot issues as well as communicate effectively to all the stakeholders.
Key Qualifications
Consistent track record of troubleshooting and resolving issues in live production environments though deep domain knowledge and implementing strategies to eliminate them.
Fast learner with excellent analytical problem solving and communication skills.
The ability to design, author, and release code in any language (Java, Swift, Javascript, Unix Scripts would be a plus)
Monitoring and log mining experience including Proficiency in using Splunk.
An ability to understand large complex systems and a passion to constantly improve environments.
Expertise with tools like Postman, Charles, XCode, Git, etc.

Experience using NoSQL solutions like MonogDB, Cassandra, etc.
Knowledge of data platforms, including but not limited to: MongoDB, Cassandra, Oracle, etc.
Description
Our team builds and manage large scale web and iOS applications and services that are used by millions of customers worldwide. We strive to provide operational excellence by ensuring the highest levels of quality, performance and availability. We are looking for a highly dedicated and motivated DevOps engineer who is a quick learner, and have passion for troubleshooting, diagnose and mitigate critical failures in high pressure situations, as well as provide hands-on technical expertise during service impacting events.
You will work very closely with the engineering team to gain deep domain knowledge of these applications and be able to tackle, research, analyze , and diagnose complicated technical issues by diving into backend systems and logging. You are also expected to develop strong cross-functional relationships with Engineering, Quality Engineering, and Site Reliability Engineering across the company. If you love designing, running systems and infrastructure that will affect millions of users then this is the place for you!
Additional Requirements
Apple is an Equal Opportunity Employer that is committed to inclusion and diversity. We also take affirmative action to offer employment and advancement opportunities to all applicants, including minorities, women, protected veterans, and individuals with disabilities. Apple will not discriminate or retaliate against applicants who inquire about, disclose, or discuss their compensation or that of other applicants.
We will ensure that individuals with disabilities are provided reasonable accommodation to participate in the job application or interview process, to perform essential job functions, and to receive other benefits and privileges of employment. Please contact us to request accommodation.
'''

assistant2= '''
Responsibilities:
Our team builds and manage large scale web and iOS applications and services that are used by millions of customers worldwide. We strive to provide operational excellence by ensuring the highest levels of quality, performance and availability. We are looking for a highly dedicated and motivated DevOps engineer who is a quick learner, and have passion for troubleshooting, diagnose and mitigate critical failures in high pressure situations, as well as provide hands-on technical expertise during service impacting events.
You will work very closely with the engineering team to gain deep domain knowledge of these applications and be able to tackle, research, analyze , and diagnose complicated technical issues by diving into backend systems and logging. You are also expected to develop strong cross-functional relationships with Engineering, Quality Engineering, and Site Reliability Engineering across the company. If you love designing, running systems and infrastructure that will affect millions of users then this is the place for you!

Requirements:
Consistent track record of troubleshooting and resolving issues in live production environments though deep domain knowledge and implementing strategies to eliminate them.
Fast learner with excellent analytical problem solving and communication skills.
The ability to design, author, and release code in any language (Java, Swift, Javascript, Unix Scripts would be a plus)
Monitoring and log mining experience including Proficiency in using Splunk.
An ability to understand large complex systems and a passion to constantly improve environments.
Expertise with tools like Postman, Charles, XCode, Git, etc.
Experience using NoSQL solutions like MonogDB, Cassandra, etc.

Knowledge of data platforms, including but not limited to: MongoDB, Cassandra, Oracle, etc.
'''

user3 = '''
Responsibilities:'''

Code

```
class llm_miner:
    def __init__(self, OPEN_AI_KEY:str) -> None:
        self.key = OPEN_AI_KEY
        self.client = OpenAI(api_key=self.key)

    def clean_text(self, x:str)->str:
        return re.sub('\s\s+',' ',x)

    def get_text(self, text:str)->str:
        try:
            # Create a chat completion using the question and context
            response = self.client.chat.completions.create(
                model="gpt-3.5-turbo",
                messages=[
                    {"role": "system", "content": system},
                    {"role": "user", "content": user1},
                    {"role": "assistant", "content": assistant1},
                    {"role": "user", "content": user2},
                    {"role": "assistant", "content": assistant2},
                    {"role": "user", "content": text+user3},
                ],
                temperature=0,
                top_p=1,
                frequency_penalty=0,
                presence_penalty=0,
            )
            return response.choices[0].message.content.strip()
        except Exception as e:
            print(e)
            time.sleep(10)
            return ""
```

c. Examples of MyCareersFuture Job Description HTML

```
<p><strong>Company background:</strong></p>
<p>Focus Computer Pte Ltd has been operating since 1989, and is an established leader in the IT hardware and software fulfilment space in Singapore. We have provided a stable environment for our employees since our inception and we believe in taking care of our employees through thick and thin. We have established Focus Digitech Pte Ltd in 2020 in order to meet the growing demands for digital transformation by our customers, and we have an opportunity to hire iOs/Android Developer to join our growing team. Your entry point into the company does not limit the potential you have to rise within the company to other roles as your aspirations and attitudes may propel you towards. We believe that paper qualifications can only tell us so much, and we believe in hiring people with the right skills and not just certifications (although we will accord due credit if you have them). So, come on and discover your full potential with us as we take Focus Digitech to the next level of leadership in digital technologies!</p>
<p><br></p>
<p><strong>Responsibilities:</strong></p>
<p>· Collaborate with cross-functional teams to gather and understand client requirements.</p>
<p>· Design, develop, and maintain iOS and Android applications using native technologies (Swift, Kotlin/Java) and Flutter.</p>
<p>· Ensure the performance, quality, and responsiveness of mobile applications.</p>
<p>· Troubleshoot and debug issues, optimizing app performance as needed.</p>
<p>· Stay updated with the latest mobile app development trends and technologies.</p>
<p>· Conduct code reviews and provide mentorship to junior developers.</p>
<p>· Collaborate with UI/UX designers to create user-friendly and visually appealing mobile interfaces.</p>
<p>· Test and validate application functionality, security, and scalability.</p>
<p>· Keep documentation up to date for all projects.</p>
<p><br></p>
<p><strong>Qualifications:</strong></p>
<p>· Bachelor's degree in Computer Science, Software Engineering, or related field.</p>
<p>· Proven experience in developing iOS and Android applications using native technologies (Swift and Kotlin/Java), Flutter.</p>
<p>· Strong knowledge of mobile app development best practices, architecture, and design patterns.</p>
<p>· Proficient in version control systems (Git).</p>
<p>· Excellent problem-solving and debugging skills.</p>
<p>· Strong communication and teamwork skills.</p>
<p>· Ability to work in a fast-paced and collaborative environment.</p>
<p>· Experience with third-party libraries and APIs integration.</p>
<p>· Familiarity with agile development methodologies is a plus.</p>
```

```
<ul>
  <li><strong>Central Area</strong></li>
  <li><strong>Monday to Friday</strong></li>
  <li><strong>Experience with building layout, blueprint, raise PO, M&amp;amp;E etc.</strong></li>
  <li><strong>Knowledge of EHS documents and Project documents.</strong></li>
</ul>
<p><strong>Job Description:</strong></p>
<ul>
  <li>Ensuring client satisfaction is a top priority.</li>
  <li>Maintain project documentation, including project plans, progress reports, and other relevant records.</li>
  <li>Ensure that project deliverables meet quality standards.</li>
  <li>Help manage project scope changes and assess their impact on timelines and resources.</li>
  <li>Engage with project stakeholders and address their concerns and requirements.</li>
  <li>Providing weekly reports as needed.</li>
</ul>
<p><strong>Requirements:</strong></p>
<ul>
  <li>At least 1 year of relevant experience.</li>
  <li>Experience with PO, DO and ERP systems.</li>
</ul>
<p>We regret to inform you that only shortlisted candidates will be notified.</p>
<p>EA License No: 15C7752</p>
<p>EA Reg. No: R22107644 | Tan He Min (Astrid)</p>
```

d. Standardizing MyCareersFuture Job Descriptions

```
jd_keywords = ['things you need to do', 'job description', 'job duties', 'responsibilities', 'job
description', 'duties', 'key responsibilities', 'job details']
jr_keywords = ['things you need to have', 'qualification', 'requirements', 'qualifications']

jobpostid_list = {}
count = 0

for index, row in df_uncleaned.iterrows():
    if row['uuid'] in uuid_list:
        has_description, has_requirement = False, False
        soup = BeautifulSoup(row['description'], 'html.parser')

        # Check for 'Job Description' in strong tag and get corresponding li items
        try:
            job_description_strong = soup.find('strong', string=lambda t: any(keyword in t.lower() for
keyword in jd_keywords))
            if job_description_strong:
                ul_tag = job_description_strong.find_next('ul')
                if ul_tag:
                    job_description_items = [li.text for li in ul_tag.find_all('li')]
                    # print("Job Description Items:", job_description_items)
                    has_description = True
        except Exception as e:
            continue

        # Check for 'Job Requirements' in strong tag and get corresponding li items
        try:
            job_requirements_strong = soup.find('strong', string=lambda t: any(keyword in t.lower() for
keyword in jr_keywords))
            if job_requirements_strong:
                ul_tag = job_requirements_strong.find_next('ul')
                if ul_tag:
                    job_requirements_items = [li.text for li in ul_tag.find_all('li')]
                    # print("Job Requirements Items:", job_requirements_items)
                    has_requirement = True
        except Exception as e:
            continue

        if has_description and has_requirement: # Only when there is job description and requirement
            jobpostid_list[row['metadata']['jobPostId']] = {'job_descriptions': job_description_items,
'job_requirements': job_requirements_items}

keys_list = list(jobpostid_list.keys())

clean_job_descriptions = []

for index, row in df_clean.iterrows():
    if row['jobPostId'] in keys_list:
        job_description_items = jobpostid_list[row['jobPostId']]['job_descriptions']
        job_requirements_items = jobpostid_list[row['jobPostId']]['job_requirements']

        clean_job_description = 'Job Description: \n'+ '\n'.join(job_description_items) + '\n\nJob
Requirements: \n'+ '\n'.join(job_requirements_items)

        clean_job_descriptions.append(clean_job_description)


    else:
        clean_job_descriptions.append(None)


df_clean['Job Description'] = clean_job_descriptions

df_clean
```

e. Job Recommendation Engine

```python
class Recommendation:
    def __init__(self,resume,openai_key=None,jobtitle=None): #model = 'all-MiniLM-L6-v2' by default
        self.resume = resume
        self.jobtitle = jobtitle
        self.openai_key = openai_key
        self.client = OpenAI(api_key=self.openai_key)
        self.file_path = os.path.join(os.getcwd(),"talenttrove/data/jd_vectordb")
        print(self.file_path)
        self.chroma_client = chromadb.PersistentClient(path=r"talenttrove/data/jd_vectordb")
        self.collection = self.chroma_client.get_or_create_collection(name="mycareersfuture_jd",
embedding_function=default_ef)


    def read_word_document(self):
        doc = Document(self.resume) ### RESUME HAS TO BE A PATH OR I/O object
        text = ''
        for paragraph in doc.paragraphs:
            text += paragraph.text + '\n'
        return text

    def get_generated_jd(self):
        text = self.read_word_document()
        try:
            # Create a chat completion using the question and context
            response = self.client.chat.completions.create(
                model="gpt-3.5-turbo",
                messages=[
                    {"role": "system", "content": "Based on the given resume above, create a suitable job
posting for this resume. The job posting must include the job description, job responsibilities, and
requirements such as qualifications and skills. Do not include the company name and location in this job
posting."},
                    {"role": "user", "content": f"Resume: {text}\n\n---\n\nJob Description:"}
                ],
                temperature=0,
                # max_tokens=max_tokens,
                top_p=1,
                frequency_penalty=0,
                presence_penalty=0,
                # stop=stop_sequence,
            )
            print(text)
            return response.choices[0].message.content.strip()
        except Exception as e:
            print(e)
            return ""

        return results
```

f. Examples of Job Title and Company Extraction Pipeline

```
text = """Subject: Nomura Application Update. Body: Dear Pratham, Congratulations again on being selected to interview for the 2024 Group Data Office Graduate Program - Singapore. Your interview is confirmed as detailed below: *Date:* ]10 November 2023 *Time:* 16:00 *Location: *Marina Bay Financial Centre *Dress Code:* Business Attire *Please note that* 1. This is the Final Round interview to be conducted at Nomura Singapore Office at Level 36, Marina Bay Financial Centre Tower 2, Singapore 018983. Please ensure you factor in traveling time to the location before selecting a slot. 2. You will have two 30-min interviews within this slot, where you will be meeting with senior members from Group Data Office (GDO) division. 3. You are recommended to dress in business attire for the interview. 4. Once the interview has been scheduled, you are not able to reschedule. 5. You will need to register at the ground floor concierge with a photo ID. Please arrive 15 mins prior to the interview scheduled time. 6. Once you reach reception at level 36, please let our receptionist know that you are looking for Jesslyn Koh. 7. If you have any questions, please email the recruiter at *XYZ@nomura.com * Additional details and the ability to re-schedule your time slot can be made directly in your Application Center . Please don't hesitate to contact us should you have additional questions or require any form of personal assistance or reasonable adjustments to be made for your interview. A member of staff will be happy to help. We look forward to meeting you soon. Kind regards, *Graduate Recruitment*"""
extractor = JobTitleCompanyNameExtractor() # FlanT5Extractor()
extractor.get_company(text),extractor.get_jobtitle(text)
('Nomura', 'Graduate Program - Singapore')
```

```
text = """Subject: ResMed Interview Invitation - Data Science Intern Body: Dear Louis,

Greetings from ResMed!

This email is to confirm your video interview for the position of Data Science Intern.


Job Description: Link

Kindly see the interview details as below:-

Date
Thursday, 15 September 2022
Time
2.00 PM
Interviewer(s)
Yang Yan (Data Science Manager, DHT - Data Science)
Zhuo Qi Lee (Lead Data Scientist, DHT)
Rakesh Prakash (Lead Data Scientist, DHT - Data Science)
Zoom Link
Meeting URL:

https://resmed.zoom.us/j/91482541719?pwd=ZmNHNCtrcGs3WFZvUExQWUg2Q1IyZz09

Meeting ID:
914 8254 1719
Passcode:
004044"""
extractor = JobTitleCompanyNameExtractor() # FlanT5Extractor()
extractor.get_company(text),extractor.get_jobtitle(text)
('ResMed', 'Data Science Intern')
```

g. ChatGPT Prompts and Outputs for Data Generation for Classifier Training

The prompts and outputs can be obtained from this chat-link:

https://chat.openai.com/share/e8b5f3a9-4593-4785-b1a5-2123dbe765da

h. Github Link for Codebase:

All code for this project can be found at Github at https://github.com/caramel2001/TalentTrove