Job Recommender System

By: Winter Blues

Problem Definition

Create a recommender system that can recommend jobs based on the resume. The system will go over the resume and then search different job descriptions that match the keywords on the resume.

Problem Motivation

- Make it easier for individuals to find jobs hiring based on their skills
- Streamline the job application process
- Introduce people to other disciplines that require their skillset

Data/Dataset Description

Data Source: Indeed

Key Information: company_name, role, location, salary_estimated, job_id, job_description,url

Filters applied:

1. Key words : data

2. Posted Date: 7 (Last 7 days)

3. Experience: Entry Level

4. Job Type: Full Time

Edge Case:

1. Jobs that are not relevant to data field: **Discard data**

2. Jobs without job description: **Discard data**

3. Job without salary: **Keep data**

Data/Dataset Description

```
{
    "job_id": "job_cc6bd88852e668e7",
    "company_name": "Treehouse Technology Group",
    "role": "Data Engineer",
    "location": "Remote",
    "salary_estimated": "Estimated $97.3K – $123K a year",
    "job_description": "\nTreehouse Technology Group is strategy and data firm providing
    The ideal candidate has a combination of a business and technical experience and will
    data models and algorithms, and develop processes and tools to monitor production sys
    Visualizations, and/or Analytics\n Experience with Python\n Experience with 3rd party
    progress with in person events quarterly where we brainstorm, roadmap, and have fun t
    "url": "https://www.indeed.com/rc/clk?jk=cc6bd88852e668e7&fccid=3211509bc4d6c792&vjs=
},
```
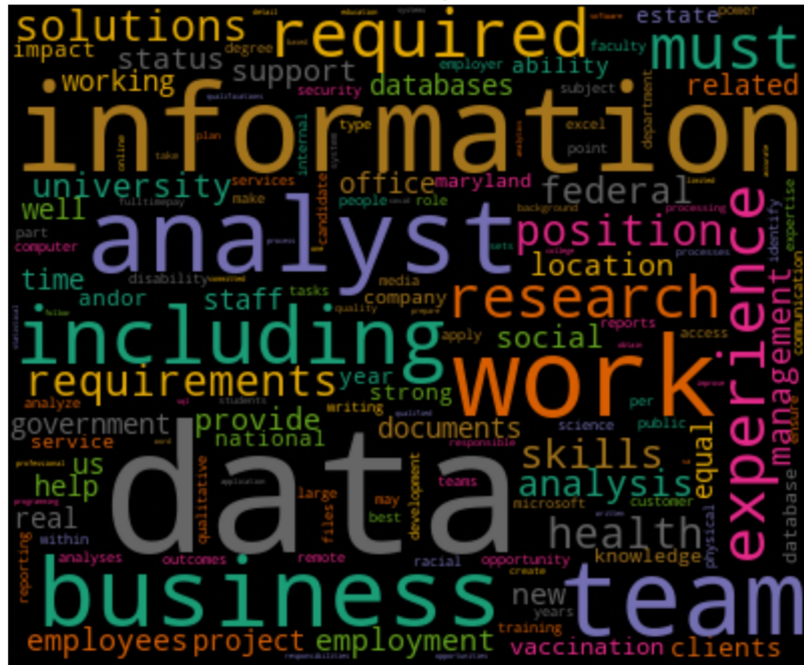
EDA Results and Inferences
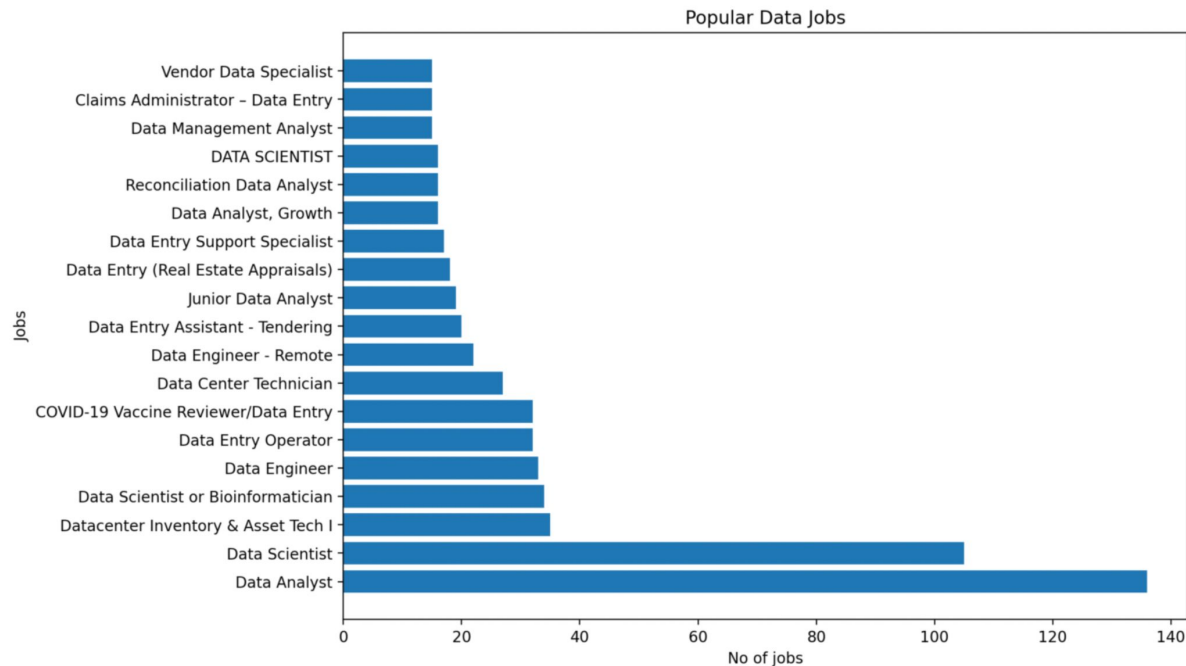
The most occuring jobs in the job data

Word clouds of Data analyst and Data scientist jobs

Model Choice

NLP(Natural language Processing) Model - Cosine similarity

1. Cleaning the job description data - removed unnecessary symbols and punctuations.

2. **Stop word removal** - removes redundant and inessential words that occur commonly across all **documents (ex. is, and, your, that, etc.), from the job description.**

3. **Resume Parsing** - extracts skills, experiences, degrees and company name from resume. This is used to extract skills from the resume which will be used to match the job description.

4. Vectorization of keywords using **TF-IDF** - assigns numerical values based on the word frequency in a document.

5. Computing **cosine similarities** with TF-IDF values and selecting the jobs with highest scores.

6. Matching resume to job description - **Cosine similarity model**.

Model Choice

Justification:

- The **ATS(Applicant tracking system)** works by identifying the resumes that are relevant to the job description and assigns them to the similar job postings.
- **Cosine similarity** gives the scores which helps in comparing the similarity of the text data, which is appropriate for our use case.
- We have compared the skills in the resume and compared them with job descriptions using similarity scores and selected the 10 most relevant jobs.

Project Pipeline

1. A web scraping program that fetches new job posts from **Indeed** daily
2. Allows users to **upload resume** as PDF format
3. Chooses 10 jobs most relevant to the resume from **S3** data lake
4. Each of the jobs that are showed in the web page will have **URL** that can take users to application website

Final Output

Top 10 relevant jobs along with their details and job posting URL, to the uploaded resume based on the skills mentioned in the resume.

Job Scraping

Team members - Shun An Chang, Harsh Nisar, Hemanth Talla

Overall Description:
Create a robust scheduled web scraping program that can get job information and store them in cloud base data lake

Progress:

- Created the web scraper for **Indeed job** board
- Created the web scraper for **linkedIn**
- Deployed the scraper on cloud through **AWS Lambda** (Stored the data to S3)
- Created **S3 API** for the web application to access the job data

Model

Team members - Chidubem Okorozo, Saad Azim, Vaibhav Patel

Overall Description:
Create a model to produce matching metric from user's resume and job description

Progress:

- Removed stop words in job description and extracted keywords from resume
- TF-IDF vectorization of skills extracted from resume
- Created KNN model to match resume to job descriptions

Dashboard

Team members - Bhargav Singuluri, Sravani Dulipalla

Overall Description:

Create a Web based Dashboard allow user to interact such as uploading resume and show users the most relative jobs for them based on their resume and EDA.

Progress:

- Integrated resume parser into the webapp to extract skills from the pdf resume file.
- Automated the integration of web scraped data into the model and EDA.
- Developed a new cosine similarity model and deployed it on webapp.
- Worked on the visual presentation of results in the webapp.

Roles & Responsibilities

Team leader: Shun An Chang
Responsibility:
- Organize the team and host the meeting
- Make sure every task done before the due date.

Team Secretary: Chidubem Okorozo
Responsibilities:
- Take notes for every meeting
- Record attendance of every meeting

Team Git manager: Sravani Dulipalla
Responsibilities:
- Manage the Git Repository
- Make sure every team member use the git properly

Team Task manager: Hemanth Talla
Responsibilities:
- Make sure every team member finish their task on time.
- Make sure every subgroup to use the task board properly.

Roles & Responsibilities - Task Completed

Job scraping

- Completed the web scraper for **Indeed** job board – Shun An Chang
- Completed the web scraper for **LinkedIn** job board - Harsh
- Created a Indeed **Web API** reader to extract job profiles in tabular format - Hemanth Talla
- Deployed the scraper on cloud through **AWS Lambda** (Stored the data to **S3**) – Shun An Chang
- Created **S3 API** for the web application to access the job data – Shun An Chang

Model

- **Stop words removal** and resume parser - Chidubem Okorozo
- **Ngrams and TF-IDF** vectorization - Saad Azim
- Matching skills in resume to job descriptions using **KNN** method - Vaibhav Patel

Dashboard

- Created a webapp with streamlit for the WIP1 presentation with EDA with preprocessed data - Bhargav Singuluri
- Integrated **resume parser** into the webapp to extract skills from the pdf resume file - Sravani Dulipalla
- Automated the **integration** of web scraped data into the model and EDA - Bhargav Singuluri
- Developed a new **cosine similarity model** and deployed it on webapp - Bhargav, Sravani
- Worked on the **visual presentation** of results in the webapp - Bhargav, Sravani

Challenges

- Setting up chrome drivers for LinkedIn Scraper
- Deploy the scraping program with AWS Lambda
- Setup S3 bucket policy - security
- Preprocessing job description - library compatible issue
- Resume parsing - config file error with spacy - library compatible issue