PROJECT REPORT

ON

HR – ANALYTICS JOB RECOMMENDATION SYSTEM

PROJECT REPORT

to be submitted by

ANJALI VERMA

B.Voc (TELECOM)

2003424

RAGINI

B.Voc (TELECOM)

2003432

DEPARTMENT OF PHYSICS & COMPUTER SCIENCE

FACULTY OF SCIENCE

DAYALBAGH EDUCATIONAL INSTITUTE

DAYALBAGH AGRA (UP)-282005

INDEX

DECLARATION

I hereby declare that the work in this project is my own except for quotations and summaries which have been duly acknowledged. The project has not been accepted for any degree and is not concurrently submitted for the award of another degree

Anjali Verma
2003424

RAGINI
2003432

ACKNOWLEDGEMENT

It gives me immense pleasure to express my deepest sense of gratitude and sincere thanks to my highly respected and esteemed guide Ms. SONIYA for her valuable guidance, encouragement, and help in completing this work. Her useful suggestions for this whole work and cooperative behaviour are sincerely acknowledged.

I am grateful to my teachers for their constant support and guidance. I am very thankful for their constant moral support and criticism which prove to be very helpful in the completion of this project.

CHAPTER-1 (INTRODUCTION)

1.1 INTRODUCTION- Company firms and recruitment organizations method several resumes each day. Going thru the resumes of these human beings manually is extraordinarily time-consuming.
In the present system the candidate has to fill each and every information regarding there resume in a manual form which takes large amount of time and then also the candidates, are not satisfied by the job which the present system prefers according to their skills.

- This project is customized as only IT or ML/AI related jobs are suggested, as dataset is related to IT based jobs.
- To improve this project, web scrapping from different websites and by pre-processing it will increase the accuracy of ML algorithm
- To overcome the problem this project will,
    - Saves the time of the candidate by providing suggested jobs on the basis of only skills provided by the candidate.
    - Help in getting the job in that company which really appreciates candidates skill and ability.
    - Saves the time of the candidate by providing percentage of how much skills present in the resume matches the job description.
    - Extract required information about candidates without having to go through each resume manually.
    - Replace slow and expensive human processing of resumes with extremely fast and cost-effective.

1.2 DEFINITION OF KEY TERMS

i. **Machine learning: -** Machine learning is a type of artificial intelligence that enables self-learning from data and then applies that learning without the need for human intervention.

ii. **Machine learning algorithms: -** Machine learning algorithms are a combination of math and logic that adjust them to perform more progressively once the input data varies. Different types of MLA's are –
   - K-nearest neighbour (knn)
   - Linear regression
   - Polynomial regression
   - Logistic regression

iii. **K- nearest neighbour:-** K-nn algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a good suite category by using k- nn algorithm. K-nn is a non-parametric algorithm, which means it does not make any assumptions about underlying data. It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.

iv. **Pyresparser: -** A simple resume parser used for extracting information from resumes.

v. **Cosine-Similarity: -** is a measure of similarity between two sequences of numbers.

vi. **CountVectorizer:** – A tool, used to transform a given text into a vector on the basis of the frequency (count) of each word that occurs in the entire text.

vii. **Python Flask: -**Flask is a web framework, it's a Python module that lets you develop web applications easily. It's has a small and easy-to-extend core: it's a microframework that doesn't include an ORM (Object Relational Manager). Flask is a web framework that provides libraries to build lightweight web applications in python. It is developed by Armin Ronacher who leads an international group of python enthusiasts (POCCO).

viii. **Python Spacy:-**spaCy is a free, open-source Python library that provides advanced capabilities to conduct natural language processing (NLP) on large volumes of text at high speed. It helps you build models and production applications that can underpin document analysis, chatbot capabilities, and all other forms of text analysis.

1.3 PROBLEM STATEMENT

The problem statement for this version of the project is, that the present resume scanning systems without machine learning are not much flexible and efficient and time saving. To overcome this problem our project saves the time of the candidate by providing suggested jobs on the basis of only skills provided by the candidate. The ML algorithm will give the best suggested job for that particular candidate and it will help in getting the job in that company which really appreciates candidates skill and ability.

1.4 OBJECTIVES OF THE STUDY

- To develop HR analytics job recommender system using machine learning system.
- To analyse the resume of student and comparing it with company's requirements.
- To select possible candidates for right job.
- To study how to use machine learning techniques for job recommendation system.

CHAPTER-2
(REVIEW OF RELATED LITERATURE)

Literature Review

2.1 Introduction- The recommender system is becoming part of every business. The business tries to increase its revenue by raising the user's interaction by recommending new items based on user preferences. We have witnessed the rise of Netflix in the entertainment domain, using their strategies to implement a recommender system into their existing ecosystem. But there has been a minimal study in the hiring field from the perspective of a job seeker. To start any research, it is quintessential to review relevant work in the domain and technology.

2.2 Recommender Systems- As discussed previously, RecSys are the system that analyses user preference history and caters them with different options of services related to the requirement. Recommender systems emerged as an independent research area in the mid-1990s. In recent years, the interest in recommender systems has dramatically increased. In the Recommendation algorithm, it classifies into four types: Content-based filtering, Collaborative filtering, Rule-based, and Hybrid approaches.

2.3 K-Nearest Neighbour (KNN) Algorithm for Machine Learning

- K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique.
- K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.
- K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.
- K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.
- K-NN is a **non-parametric algorithm**, which means it does not make any assumption on underlying data.
- It is also called a **lazy learner algorithm** because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.
- KNN algorithm at the training phase just stores the dataset and when it gets new data, and then it classifies that data into a category that is much similar to the new data.

- **Example:** Suppose, we have an image of a creature that looks similar to cat and dog, but we want to know either it is a cat or dog. So for this identification, we can use the KNN algorithm, as it works on a similarity measure. Our KNN model will find the similar features of the new data set to the cats and dogs images and based on the most similar features it will put it in either cat or dog category.

2.4 Cosine similarity: - Cosine similarity is also a measure to find similarity between two sets of non-zero vector. It is a weighted vector space model utilized in the process of information retrieval. The similarity is measured by using Euclidean cosine rule, i.e., by taking inner product space of two non-zero vector that measures the cosine of the angle between the two vectors. If the angle between two vectors is 0deg, then the cosine of 0 is 1; Meaning that the two non-zero vectors are similar to each other.

Cosine Similarity (A, B) = cos(θ) = n ∑ i=1 Ai ∗Bi r n ∑ i=1 A 2 i ∗ r n ∑ i=1 B 2 i

We can also compute cosine distance by using below equation,

Cosine Distance = 1− (Cosine Similarity)

CHAPTER-3

Methodology

Text data is one of the principal kinds of data. As extracting of data from the job board is done, tools are used to scrape the data from the website. These text data will be in an unstructured format, so pre-processing of data before going to the modeling stages is necessary. Once the extracted data is processed and transformed into feature, the project can further go ahead utilize those features as input to the KNN classifier. Then it feeds the user skills and job description to a similarity measure algorithm to identify the similarity between the job and user. Once the similarity matches, it will be recommending the job that has high similarity, and display top-n scored jobs to the users' dashboard, which is built on top of the Flask web framework. The feature creation and algorithms used in this project are explained below in details:

3.1 Web scraping - rvest

For any study to be performed, the main concern is to collect the data that is useful to the study. If the data is available on the internet in an HTML format on a particular website, then a traditional way to collect the data would be hard, as it would be time-consuming. The data that are available on the internet will be in forms tables, comments, articles, job listing which are embedded in different HTML tags. Gathering such data would not be an easy task considering the volume of it. So we rely on the method such as web scraping. This technique came into existence right around the year, where the internet was introduced to the world.

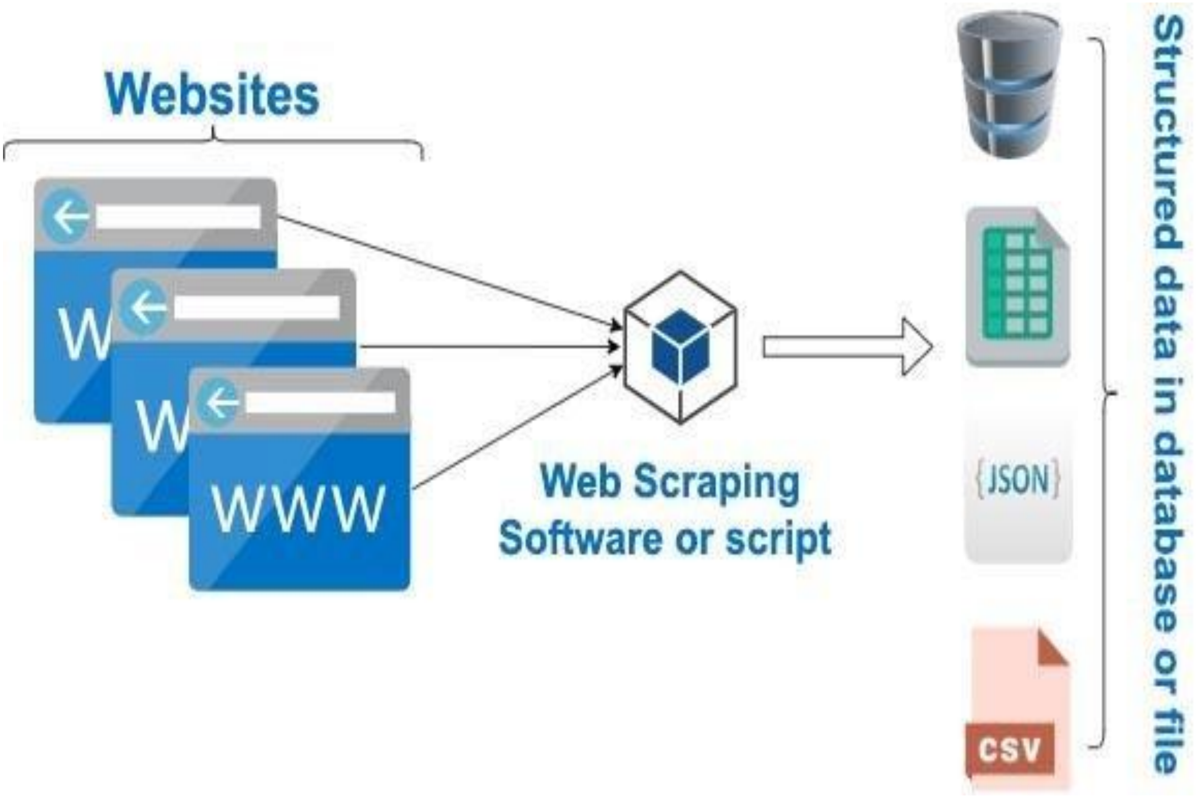
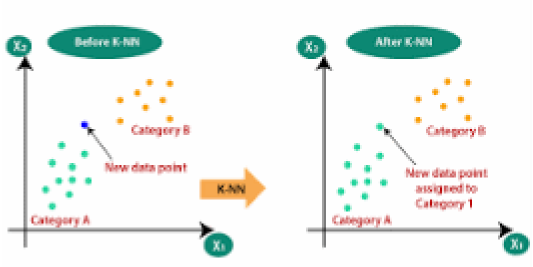


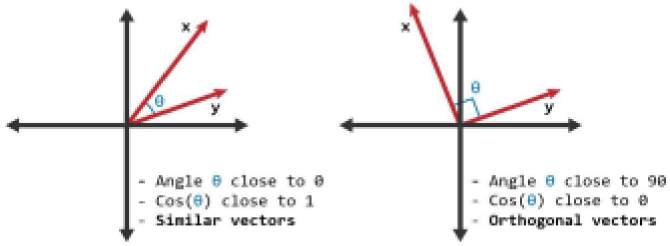
Fig. 3.1 Web Scrapping overview

Web scraping is the technique used to collect or extract information from the web sources and store it locally for the further analysis of the user’s choice. Web scraping is also termed as screen scraping, web data extraction, or even web harvesting. Data in HTML language can be viewed through a web browser. Every website has its own structure, so the method of web scrapping is hard to generalize for every website. So we rely on automating or creating a web crawler using python language. Python has a package called Beautiful Soup, which is used to request the HTML data and parse it using HTML nodes available in the file. As websites try to put a restriction on the client system to avoid web scraping, the code which was intended to automate the extraction of the data would get blocked by a site. R programming also supports developers with web scraping packages such as rcurl and rvest. rcurl is the base web scraping package provided by the R, whereas rvest is also a package for web scraping developed by tidyverse inspired by beautiful Soup. Both packages load HTML to an object, but the difference with rvest is it is not just a web parser, but it can connect to a web page, scrape and parse HTML in a single package.

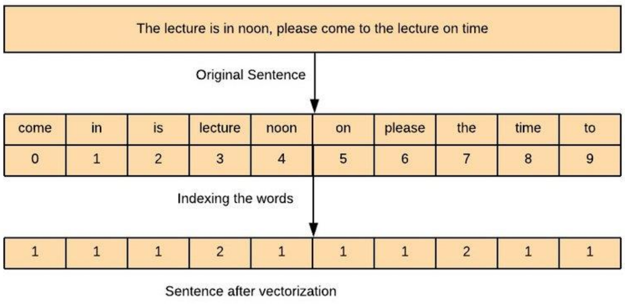
HR Analytics, is the tool for HR to suggest the top 'n' jobs from the dataset based on the skills of the person. There are two versions of this tool. In the first version, candidate will be manually providing the skills as the input. Both have there UI based on their input type. Pyreparser is used for extracting the skills from the resume.
In the second version, pyresparser is used for extracting the skills from the resume in .pdf format. For matching the job description and the skills, Cosine-similarity algorithm is used . At last deployed both the version using Flask and rendering is done using simple html templates.



K Nearest Neighbour Algorithm



Cosine - Similarity



The lecture is in noon, please come to the lecture on time

Original Sentence

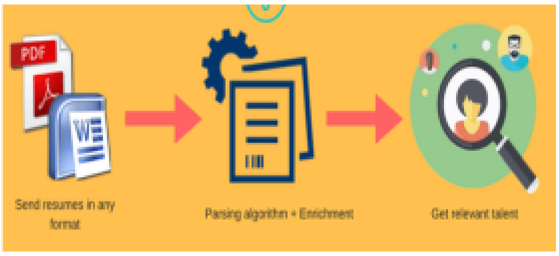
come	in	is	lecture	noon	on	please	the	time	to
0	1	2	3	4	5	6	7	8	9

Indexing the words

1	1	1	2	1	1	1	2	1	1

Sentence after vectorization

Count Vectorizer



Resume Parsing

Process Model

The process model is a series of steps, concise descriptions, and decisions involved to complete the project implementation. To finish the project within the time given, the flow of the project needs to be followed. The framework below shows how the overall flow of this project recommends jobs to different candidates.

The framework is implemented by performing the following steps accordingly –

1. Data collection using Webscraping

2. Data Cleaning

3. Feature extraction/engineering

4. Model building

5. Website making – Python Flask

6. Website deployment – future work

Follow the steps to implement the project:

Step 1 – Importing python libraries

First step is to import pandas, sklearn, and numpy libraries to read the .csv file and to make the dataset in a readable format. Sklearn is imported for machine learning algorithms.

Step 2 – Data cleaning process

In this stage, the nan values and unnamed columns are removed. Df.info() command is used to check what data is to be cleaned. So, this function shows that there are 7 columns in the dataset and 2 columns i.e., unnamed 1 & test should be removed as it contains nan values.

Step 3 – Data preprocessing/Textual preprocessing

In this stage, the following steps are performed:

• Lower case • Tokenization • Removing special characters • Removing stop words and punctuation

First, the data in the dataset is converted into lowercase letters. Then, the lowered case data is converted into smaller tokens (breaking longer sentences into small words). After then, special characters are removed. Next, stop words are removed, which are those words in the english language that are used in sentence formation but have no contribution to the actual meaning of the sentence. Eg., is, of, the. After that punctuation mark is removed. At the end, stemming is performed (removing those words which have similar meanings), for ex., dance, dancing, danced to dance, playing to play.

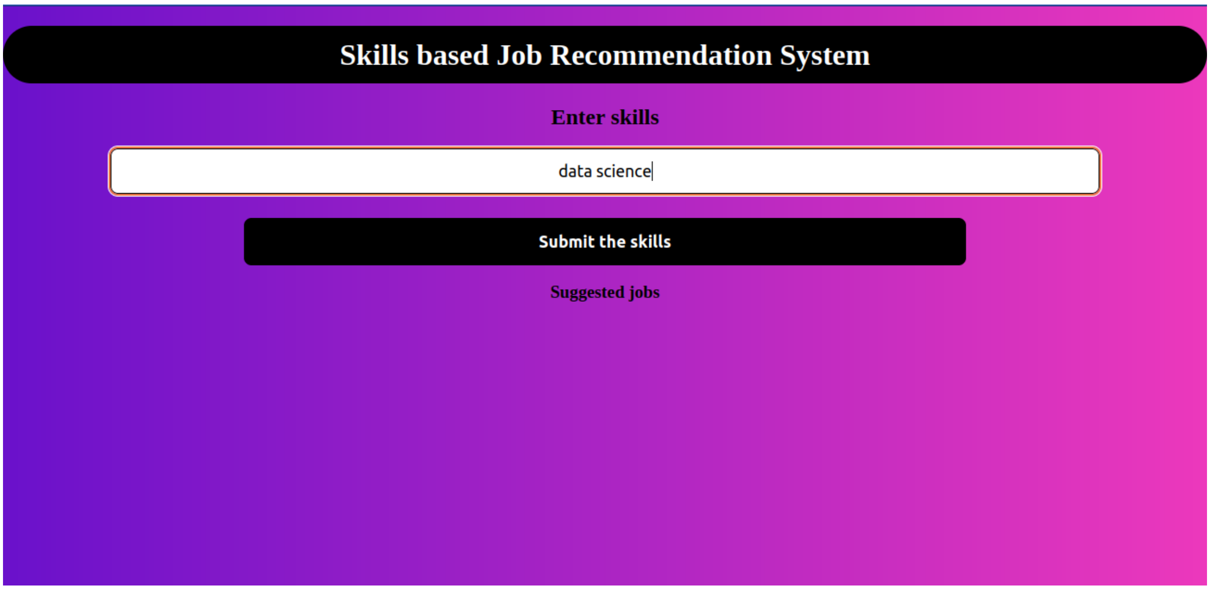
Step 4 – Model building

In this stage, for model building, the data is converted into an integer or numerical format or by performing vectorization in the data. This can be done by using count vectorization(used here), or by selecting more occurrences of frequent words in the job description and then converting them into numbers, etc.

Step 5 – Website Building

In this stage, the website is created using Python Flask Web Framework. All the above steps (in terms of code) are repeated for real time job prediction whenever the user enter skills(Vesion 1) or upload his/her resume in .pdf format.

Output

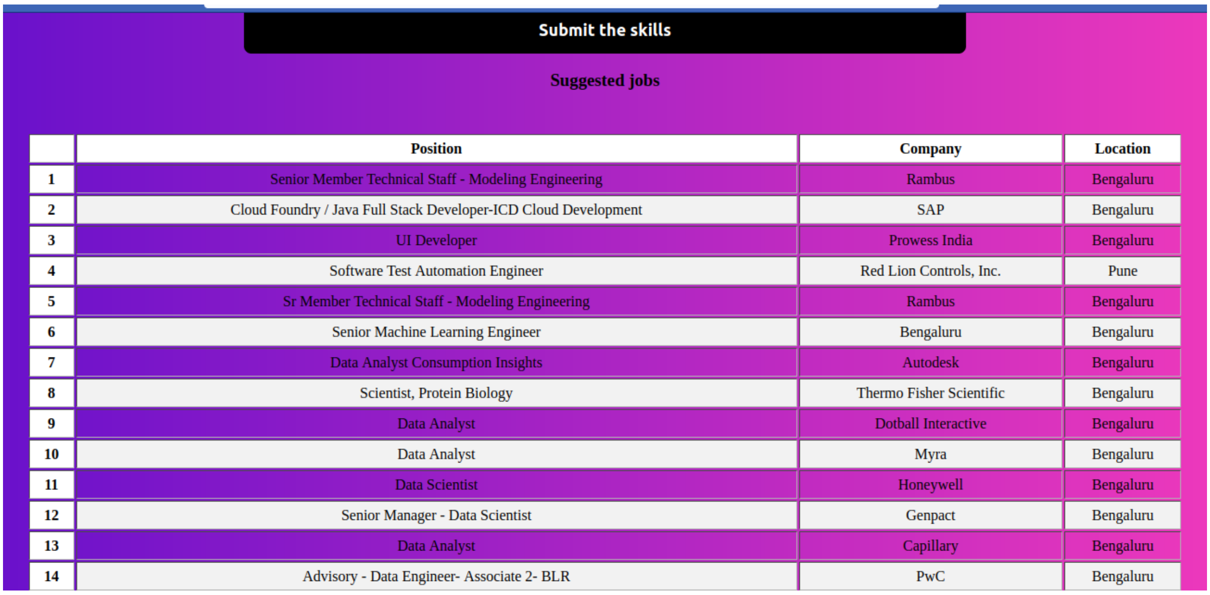


Skills based Job Recommendation System

Enter skills

data science

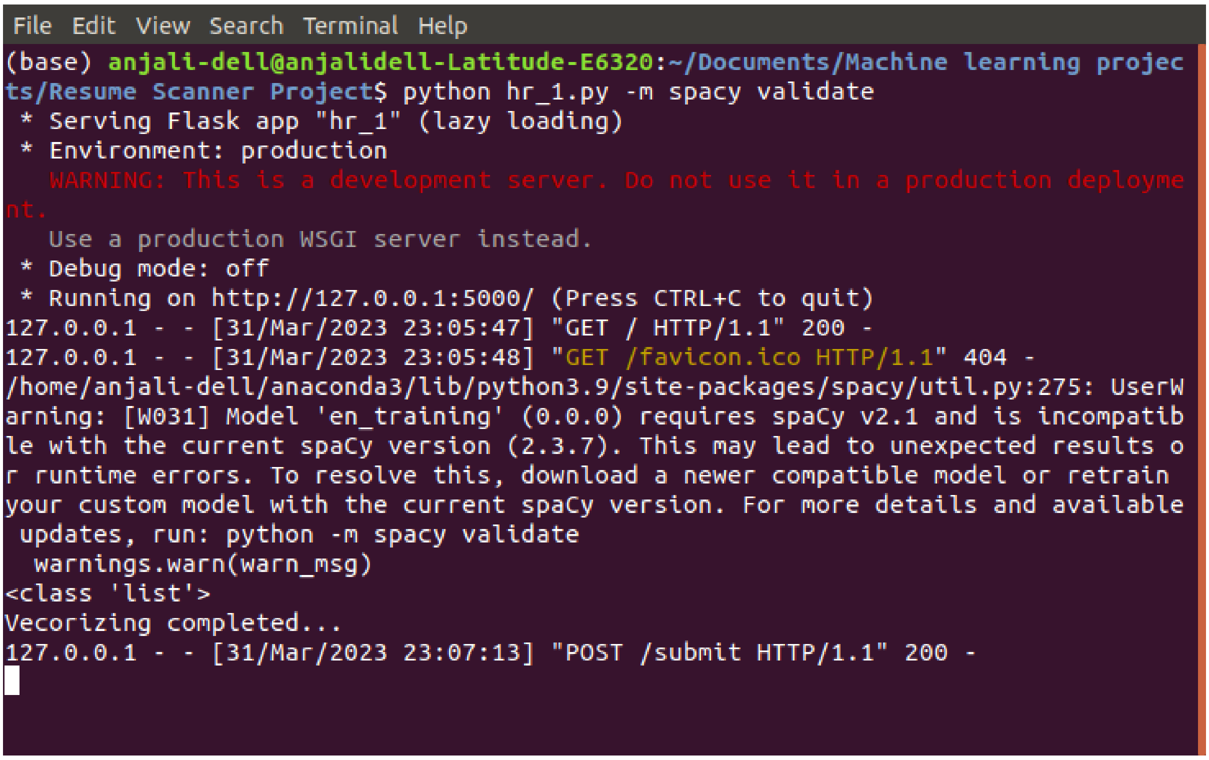
Suggested jobs



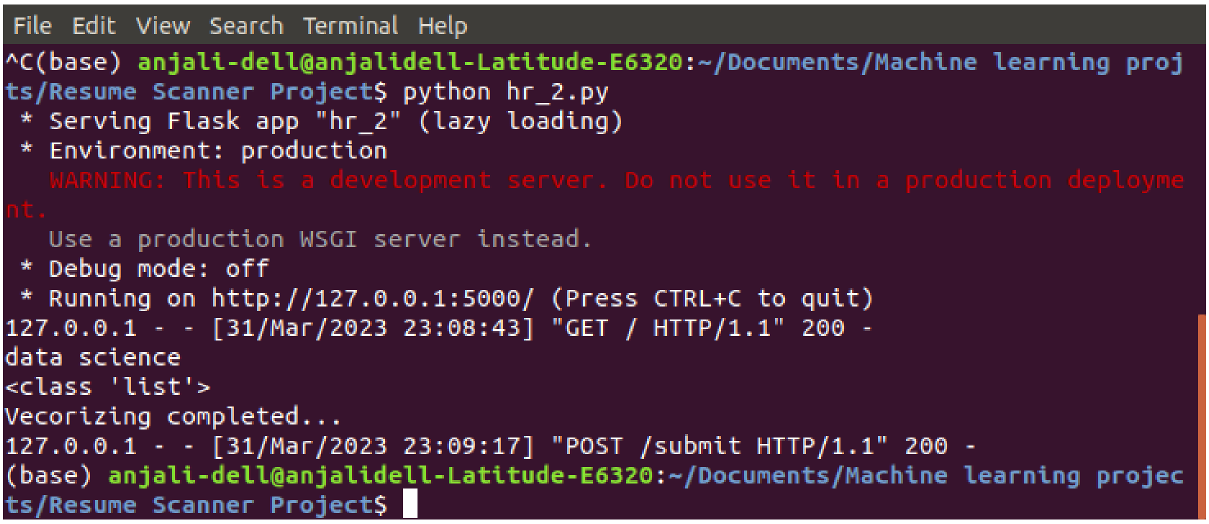
Suggested jobs

	Position	Company	Location
1	Senior Member Technical Staff - Modeling Engineering	Rambus	Bengaluru
2	Cloud Foundry / Java Full Stack Developer-ICD Cloud Development	SAP	Bengaluru
3	UI Developer	Prowess India	Bengaluru
4	Software Test Automation Engineer	Red Lion Controls, Inc.	Pune
5	Sr Member Technical Staff - Modeling Engineering	Rambus	Bengaluru
6	Senior Machine Learning Engineer	Bengaluru	Bengaluru
7	Data Analyst Consumption Insights	Autodesk	Bengaluru
8	Scientist, Protein Biology	Thermo Fisher Scientific	Bengaluru
9	Data Analyst	Dotball Interactive	Bengaluru
10	Data Analyst	Myra	Bengaluru
11	Data Scientist	Honeywell	Bengaluru
12	Senior Manager - Data Scientist	Genpact	Bengaluru
13	Data Analyst	Capillary	Bengaluru
14	Advisory - Data Engineer- Associate 2- BLR	PwC	Bengaluru

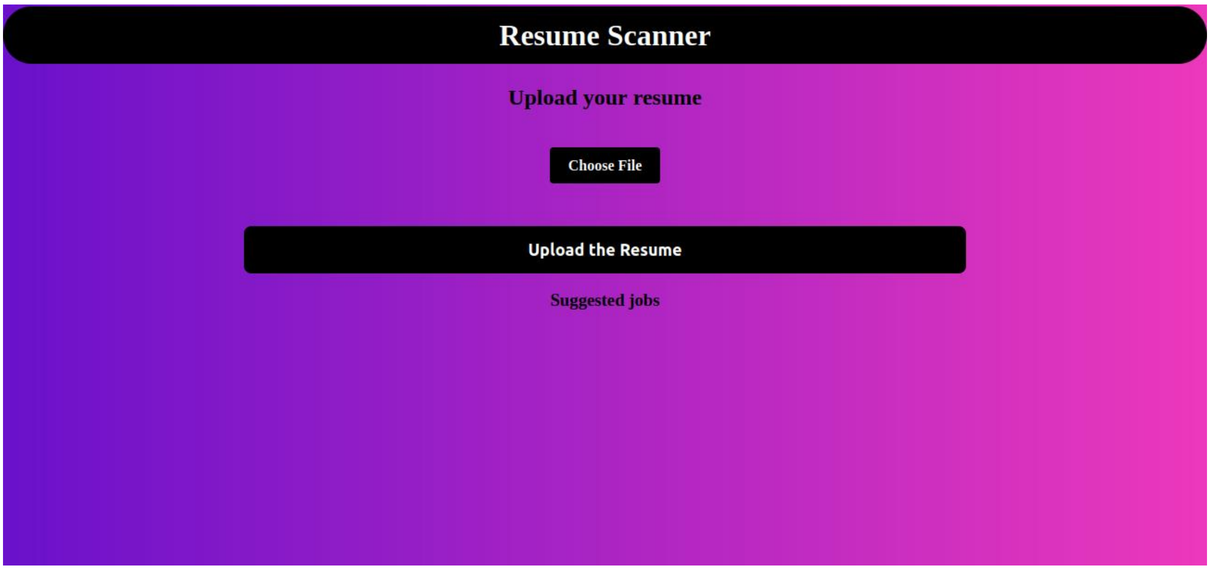
Version 1 – Skill based job recommender

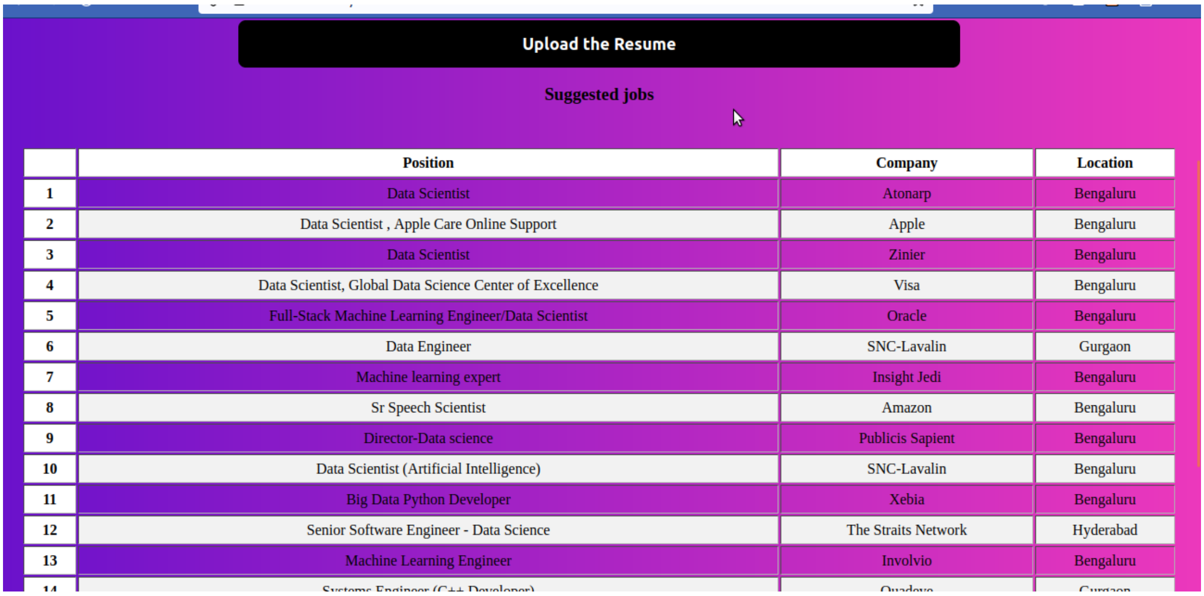


```
(base) anjali-dell@anjalidell-Latitude-E6320:~/Documents/Machine learning projec
ts/Resume Scanner Project$ python hr_1.py -m spacy validate
 * Serving Flask app "hr_1" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployme
nt.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
127.0.0.1 - - [31/Mar/2023 23:05:47] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [31/Mar/2023 23:05:48] "GET /favicon.ico HTTP/1.1" 404 -
/home/anjali-dell/anaconda3/lib/python3.9/site-packages/spacy/util.py:275: UserW
arning: [W031] Model 'en_training' (0.0.0) requires spaCy v2.1 and is incompatib
le with the current spaCy version (2.3.7). This may lead to unexpected results o
r runtime errors. To resolve this, download a newer compatible model or retrain
your custom model with the current spaCy version. For more details and available
 updates, run: python -m spacy validate
  warnings.warn(warn_msg)
<class 'list'>
Vecorizing completed...
127.0.0.1 - - [31/Mar/2023 23:07:13] "POST /submit HTTP/1.1" 200 -
```



```
^C(base) anjali-dell@anjalidell-Latitude-E6320:~/Documents/Machine learning proj
ts/Resume Scanner Project$ python hr_2.py
 * Serving Flask app "hr_2" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployme
nt.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
127.0.0.1 - - [31/Mar/2023 23:08:43] "GET / HTTP/1.1" 200 -
data science
<class 'list'>
Vecorizing completed...
127.0.0.1 - - [31/Mar/2023 23:09:17] "POST /submit HTTP/1.1" 200 -
(base) anjali-dell@anjalidell-Latitude-E6320:~/Documents/Machine learning projec
ts/Resume Scanner Project$
```





	Position	Company	Location
1	Data Scientist	Atonarp	Bengaluru
2	Data Scientist , Apple Care Online Support	Apple	Bengaluru
3	Data Scientist	Zinier	Bengaluru
4	Data Scientist, Global Data Science Center of Excellence	Visa	Bengaluru
5	Full-Stack Machine Learning Engineer/Data Scientist	Oracle	Bengaluru
6	Data Engineer	SNC-Lavalin	Gurgaon
7	Machine learning expert	Insight Jedi	Bengaluru
8	Sr Speech Scientist	Amazon	Bengaluru
9	Director-Data science	Publicis Sapient	Bengaluru
10	Data Scientist (Artificial Intelligence)	SNC-Lavalin	Bengaluru
11	Big Data Python Developer	Xebia	Bengaluru
12	Senior Software Engineer - Data Science	The Straits Network	Hyderabad
13	Machine Learning Engineer	Involvio	Bengaluru
14	Systems Engineer (C++ Developer)	Quadeye	Gurgaon

Version 2 – Resume Scanner

Implementation & Coding

The code for website deployment is written in python. For local deployment Python Flask web framework is used. Thus, this project contains two versions for now, the first version contains hr_1.py python file and the second version contains hr_2.py python.

For HTML templates, hr_1.py file is internally connected to the model.html HTML file, and hr_2.py file is internally connected to the new_model.html HTML file, using the following code.

```
#hr_1.py
@app.route('/')
def hello():
        return render_template("model.html")
#hr_2.py
@app.route('/')
def hello():
        return render_template("new_model.html")
```

For final rendering of code in the website is done by following these steps:
(Version 1 – Skill based job recommender)
Step 1 – Open the folder in the Terminal where the hr_1.py file is present.
Step 2 – Run the python file by writing given code:

```
python hr_2.py
```

This will give a real time sever link which is a development server and is not used as the production deployment.

Step 3 – Click on the URL by pressing CTRL button.
Step 4 – It will open the website of Version 1st – Skill based job recommender.

For final rendering of code in the website is done by following these steps:
(Version 2-Resume Scanner)
Step 1 – Open the folder in the Terminal where the hr_1.py file is present.
Step 2 – Run the python file by writing given code:

```
python hr_1.py -m spacy validate
```

This will give a real time sever link which is a development server and is not used as the production deployment.

Step 3 – Click on the URL by pressing CTRL button.
Step 4 – It will open the website of Version 2nd – Resume Scanner.

CHAPTER-4 (Future work and Discussion)

4.1 Future work

Based on the current study, the recommendation system works as two different versions and it does not rendered on the same website. Also, the created dataset contains only IT, ML, AI, Data Science, related jobs and hence the ML algorithm will only suggest IT related jobs. This will limit the candidates skills which are related to non-IT skills.

Since currently, the dataset consists only 1920 jobs the classifier may not be able to bundle the groups having similar features. To improve more webscraping should be done to increase the accuracy.

The corpus provides general information about the word and similar words around it, It is possible to create a better recommendation by creating a corpus related to the IT skills, terminology, Job domain and jargon of the industry. By using such corpus specific to the hiring domain, the recommendation could be better when analyzing implicit text data in the job description. It can be categorized in a better way.

As Glassdoor is currently working on data that has no interaction, a study needs to be conducted on the data that has previous interaction in the hiring domain. This would allow us to dynamically keep recommending new jobs based on user's change in preferences. There is a recommender system in the hiring domain from LinkedIn but not in the perspective of a job seeker but from the perspective of a recruiter.

4.2. Discussion

This study on recommender system in the field of the hiring domain concentrates on analysing the skills required for the job, to which domain user fall into; using this as a parameter to compute the similarity between available position and user. The available recommender system in the domain of news or the field of entertainment relies on user interaction. Interaction such as ratings provided by the user on a particular item, to make an item recommendation to a user but this concept of ratings and predicting the likelihood of a user to choose an item would be incorrect when it is viewed in the perspective of job domain or recruiter. Implementation of a recommender system that is based on ratings, number of views and popularity of an item in the job domain would allow the user to apply most of the job that he sees online. However, this would hamper the process of hiring due to the clogging of profile at the recruiter end. In this study, we recommend the job that is similar to the user profile by analysing the user preference of the user using content-based filtering. Using this process of recommendation would efficiently make the user apply only to the jobs that he might be suited to, instead of applying to most of the jobs that are available in the system. This recommender system would ease the burden of a recruiter by reducing the number of irrelevant applicants.

CHAPTER- 5

1. Conclusion

Therefore, We conclude that job recommendation system with analysis of job description to recommend a job based on user's skills and preferences presents itself as worthy Recsys model in recommending open position to the job seekers when looking for a new positions.

REFERENCES

https://youtu.be/X83cDfwtFpw

https://github.com/pik1989/HRAnalytics-Project

https://www.researchgate.net/publication/361772014_RESUME_PARSER

https://core.ac.uk/download/pdf/55305289.pdf

https://www.researchgate.net/publication/340700219_A_Machine_Learning_approach_for_automation_of_Resume_Recommendation_system

https://www.analyticsvidhya.com/blog/2021/06/resume-screening-with-natural-language-processing-in-python/ --Resume Screening with Natural Language Processing in Python using dataset

https://randerson112358.medium.com/resume-scanner-2c30f5baf92c

https://oindrilasen.com/2021/05/build-resume-scanner-using-python-nlp/

https://github.com/Vignesh0404/Resume-Scanner

https://esource.dbs.ie/bitstream/handle/10788/4254/msc_jeevankrishna_2020.pdf?sequence=1&isAllowed=y