Project Phase #1

Team Member 1: DHIVYASHREE SIVAPRAKASAM (UBIT – dsivapra, Person number - 50560688)

Team Member 2: SHALINI ANANTHAVEL JAYALAKSHMI (UBIT – ananthav, Person number - 50560497)

Team Member 3: GANESH PRABAKARAN (UBIT – ganeshpr, Person number - 50560751)

1. PROBLEM STATEMENT:

The exponential growth of online job portals and professional networking platforms has revolutionized the recruitment process by providing individuals with convenient access to job listings and networking opportunities. However, the sheer volume of available job postings often results in information overload, making it difficult for job seekers to identify relevant opportunities. Moreover, traditional job search methods often lack personalization and fail to consider individual preferences, skills, and career aspirations. Consequently, there is a pressing need for an intelligent system that can analyze user data and provide personalized job recommendations tailored to the unique profile of each user.

The project seeks to solve the problems of information overload, lack of personalization and inefficiency in the job-hunting process by developing a personalized job recommendation system that leverages user data and machine learning algorithms to deliver tailored job suggestions.

Problems:

Inefficient Search: Individuals spend a considerable amount of time sifting through irrelevant job postings, leading to frustration and wasted effort.

Skills-Gap Mismatch: Companies struggle to find candidates with the specific skills and experience they require, leading to longer hiring times and potentially unfilled positions.

Underutilized Talent: Individuals with valuable skills and experiences may be overlooked due to a lack of visibility in the traditional job search process.

Significance:

- The evolution of a system offering personalized job recommendations promises to significantly impact both the individuals seeking employment and those tasked with recruiting them. For job

seekers, this technology has the potential to streamline the search process by surfacing relevant opportunities that closely align with their unique skillset, aspirations, and career goals. This not only translates to a more efficient and less time-consuming experience, but also increases the chances of finding a truly fulfilling job match.

- Similarly, recruiters stand to benefit by gaining access to a curated pool of qualified candidates whose profiles closely resemble the requirements of open positions. By leveraging this system, the recruitment process can become significantly more efficient and effective, leading to more targeted outreach and ultimately, a higher success rate in filling vacant positions.

Why crucial contribution?

By leveraging machine learning techniques and user profiling, the system can provide personalized recommendations that go beyond traditional keyword-based searches, leading to better job outcomes for both job seekers and recruiters. This contribution is crucial in addressing the growing challenges associated with talent acquisition and retention in today's dynamic job market, ultimately facilitating better alignment between individuals and employment opportunities.

2. DATA SOURCE:

Kaggle : https://www.kaggle.com/datasets/asaniczka/1-3m-linkedin-jobs-and-skills-2024

LinkedIn is a widely used professional networking platform that hosts millions of job postings. This dataset contains 1.3 million job listings scraped from LinkedIn in the year 2024.

This is the same master dataset that powers https://skillexplorer.asaniczka.com/

For our problem statement we require 2 out of 3 csv files from the dataset.

Dataset 1: **linkedin_job_postings.csv**

A comprehensive dataset with 527501 rows × 14 columns job postings scraped from LinkedIn in 2024. Includes job details, company information, locations, and more.
Dataset details:

Column Name	job_link	todaylast_processed	got_summary	got_ner	is_being_worked	job_title	company	job_location	first_seen	search_city
Description	URL link to the job posting on LinkedIn. (type:str)	Timestamp indicating the last time the job posting was processed. (type:datetime)	Indicates whether the job summary was successfully extracted or not. (type:bool)	Indicates whether Named Entity Recognition (NER) was performed on the job posting or not. (type:bool)	Indicates if the job posting is currently being worked on or not. (type:bool)	Title of the job listing. (type:str)	Company name offering the job position. (type:str)	Location of the job position. (type:str)	Timestamp indicating when the job posting was first seen. (type:datetime)	Cityused as a search criterion for collecting the job postings. (type:str)

Dataset 2: **job_skills.csv** Contains 1296381 rows × 2 columns skills extracted via NER from job summary.

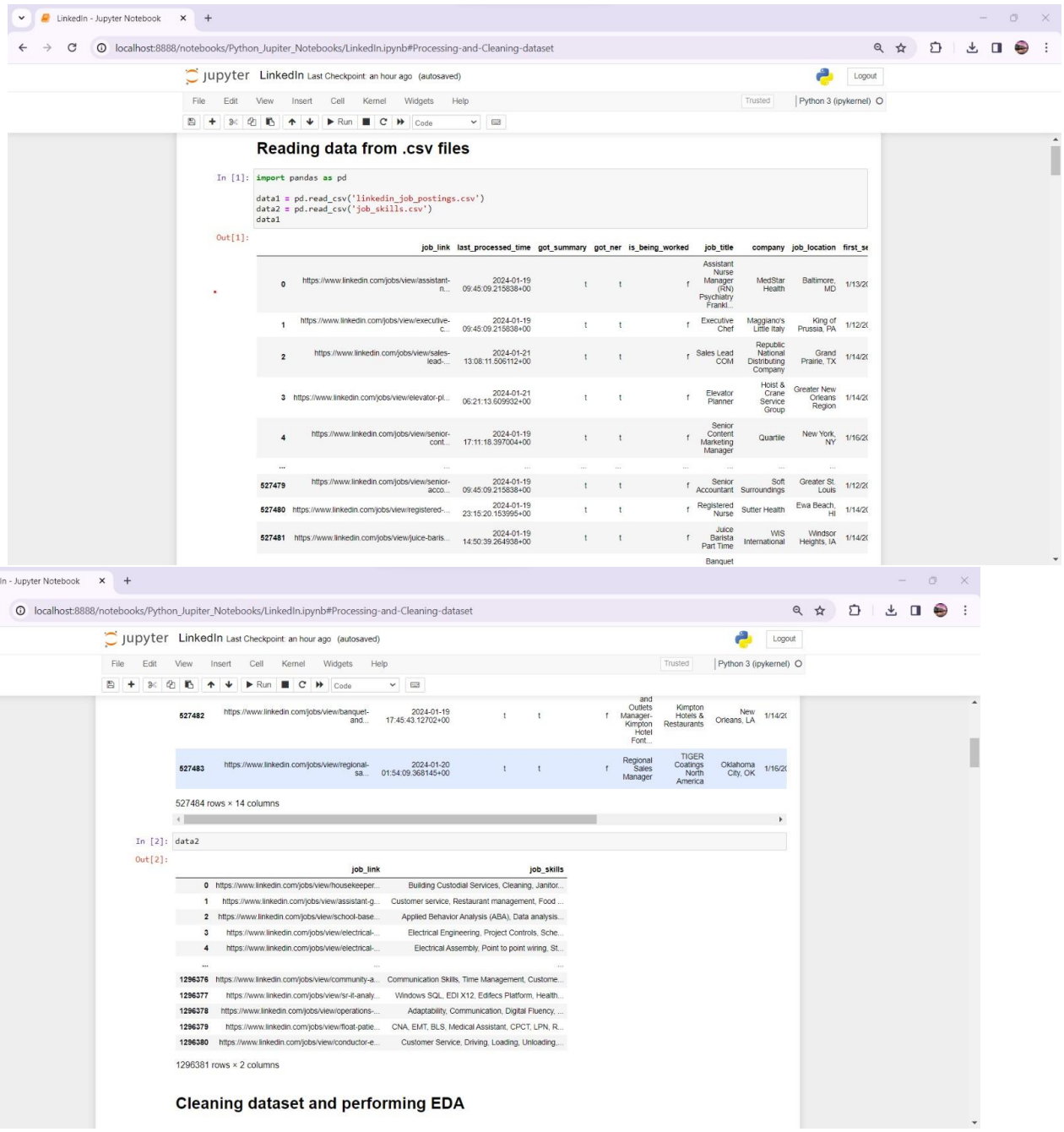
Column Name	job_link	text_formatjob_skills
Description	Foreign Key	List of skills.

3. DATA CLEANING:

The two data files (linkedin_job_postings.csv & job_skills.csv) are loaded using panda functions and assigned to two variables data1, data2 respectively.

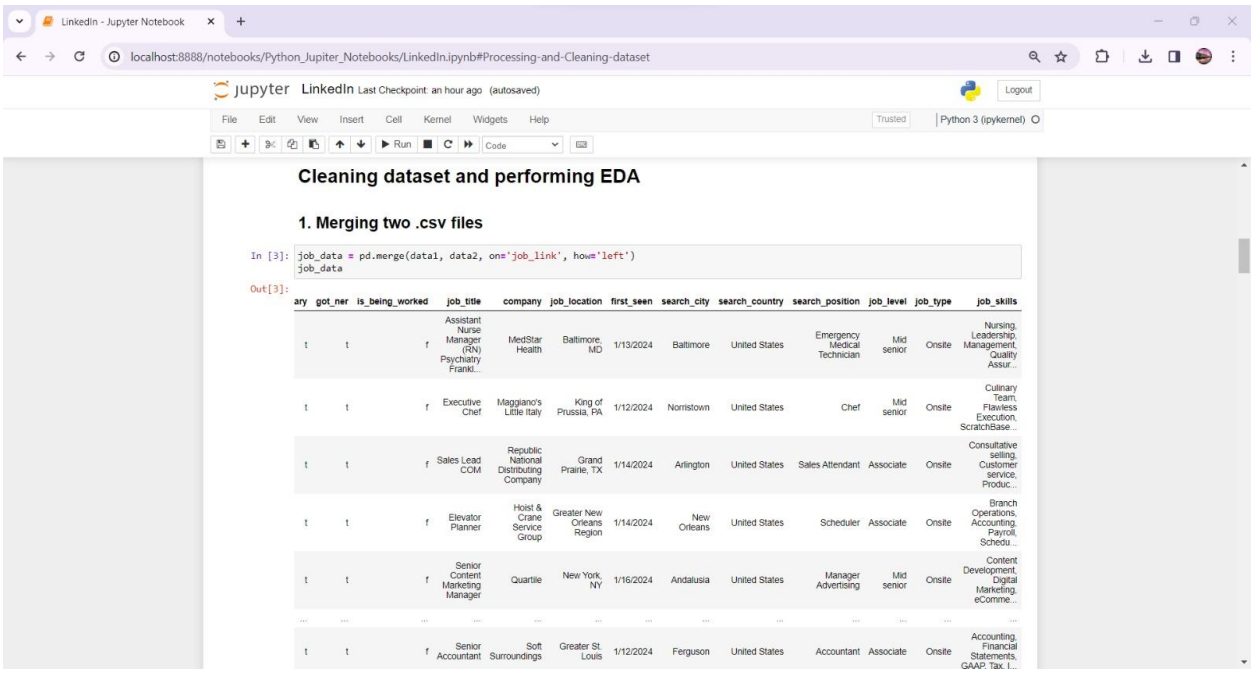
BEFORE CLEANING:

No. of rows	527, 501
No. of columns	14



3.1 MERGING TWO .csv FILES

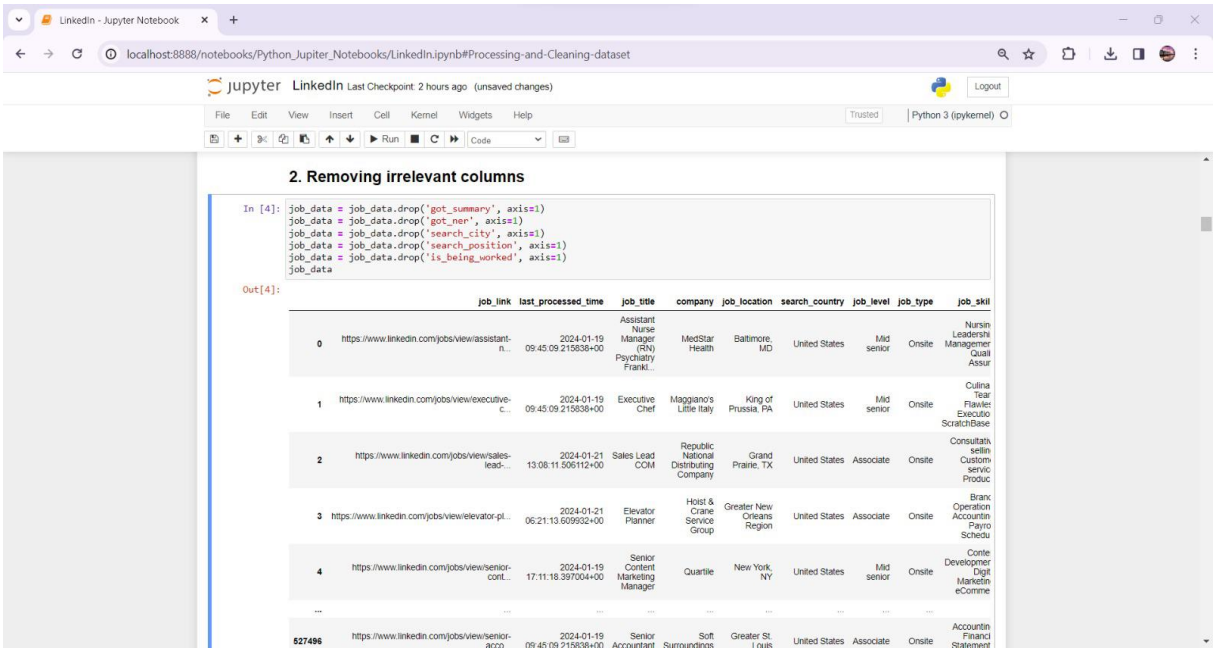
The two source files are merged together into a single csv file using the built-in pandas method. Left join is used to merge the two files with common column ‘job_link’.



3.2 REMOVING IRRELEVANT COLUMNS

- Removing irrelevant data leads to a cleaner dataset that is more relevant to the specific analysis needs, promoting data accuracy and reducing the risk of misleading conclusions.
- We are dropping the below columns, since for job recommendations valuesin these columns do not affect the prediction of jobs.

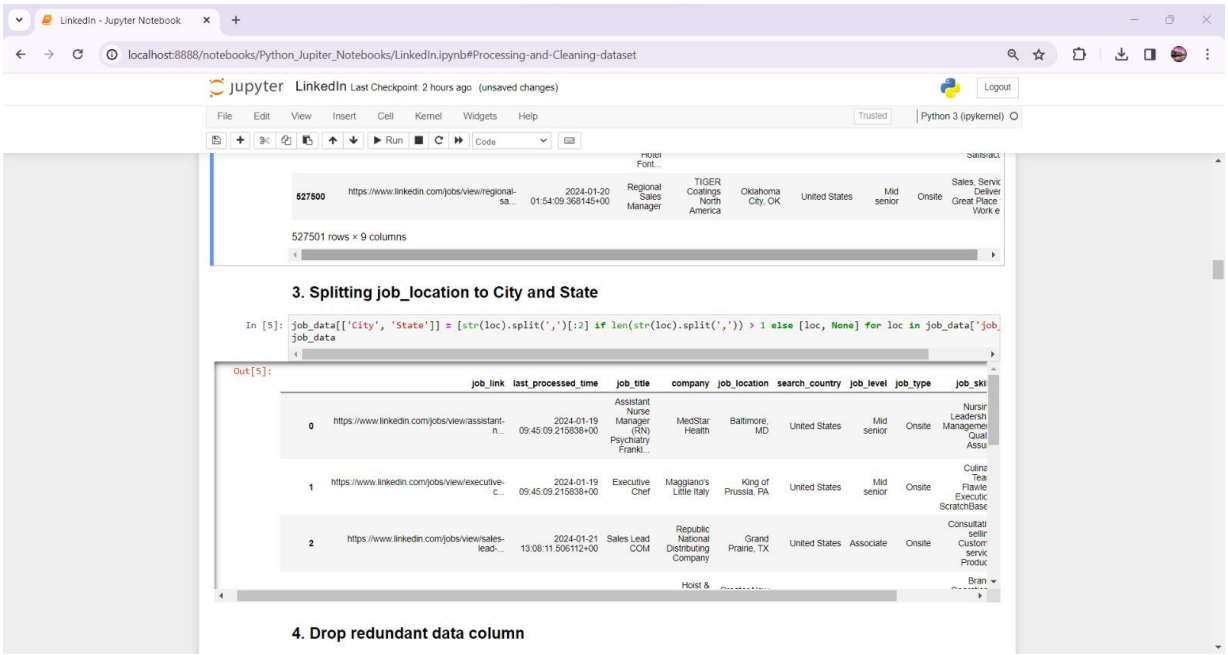
{‘got_summary’,’got_ner’,’first_seen’,’search_city’,’search_position’,’is_being_worked’} using the built-in pandas method(data_drop).



3.3 SPLITTING ‘job_location’ TO ‘city’ AND ‘state’

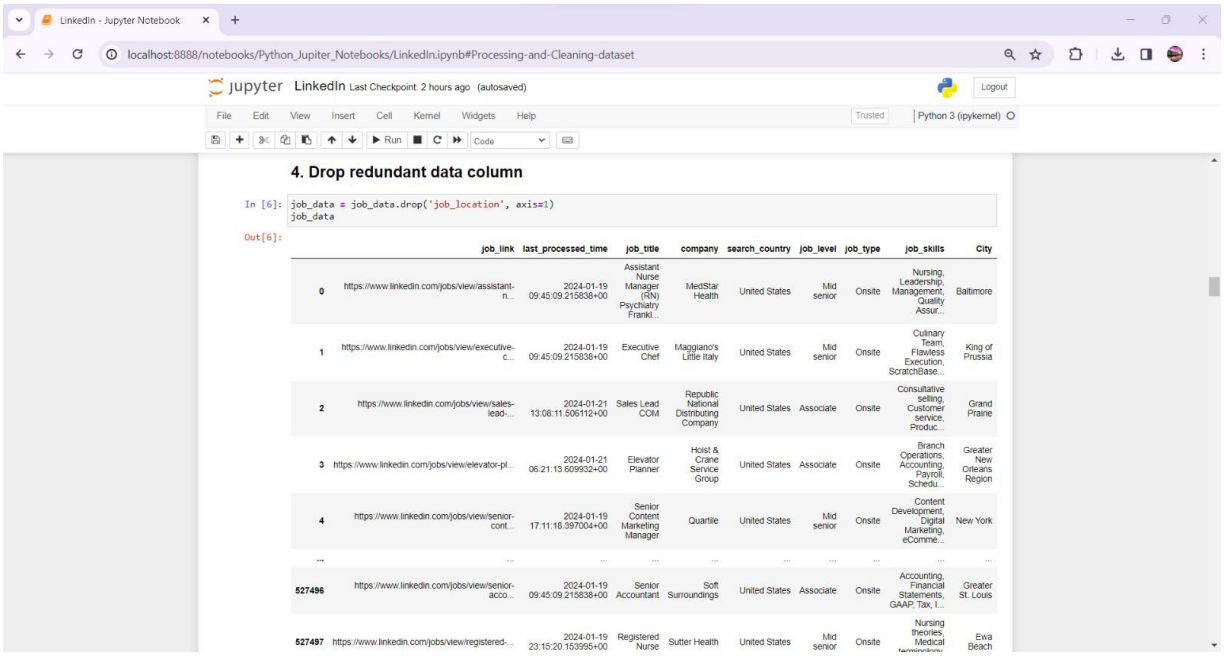
To better understand the data based on the location we are splitting the column ‘job_location’ into two columns named ‘city’ & ‘state’.

An example of job_location row is ‘San Diego, CA’. Using split method, we assign the text before the first comma to column City and the latter to State.



3.4 DROP REDUNDANT DATA COLUMN

Since, we have city and state two columns for location. ‘job_location’ column is redundant and hence we are dropping the column ‘job_location’.



4. Drop redundant data column

```
In [6]: job_data = job_data.drop('job_location', axis=1)
        job_data
```

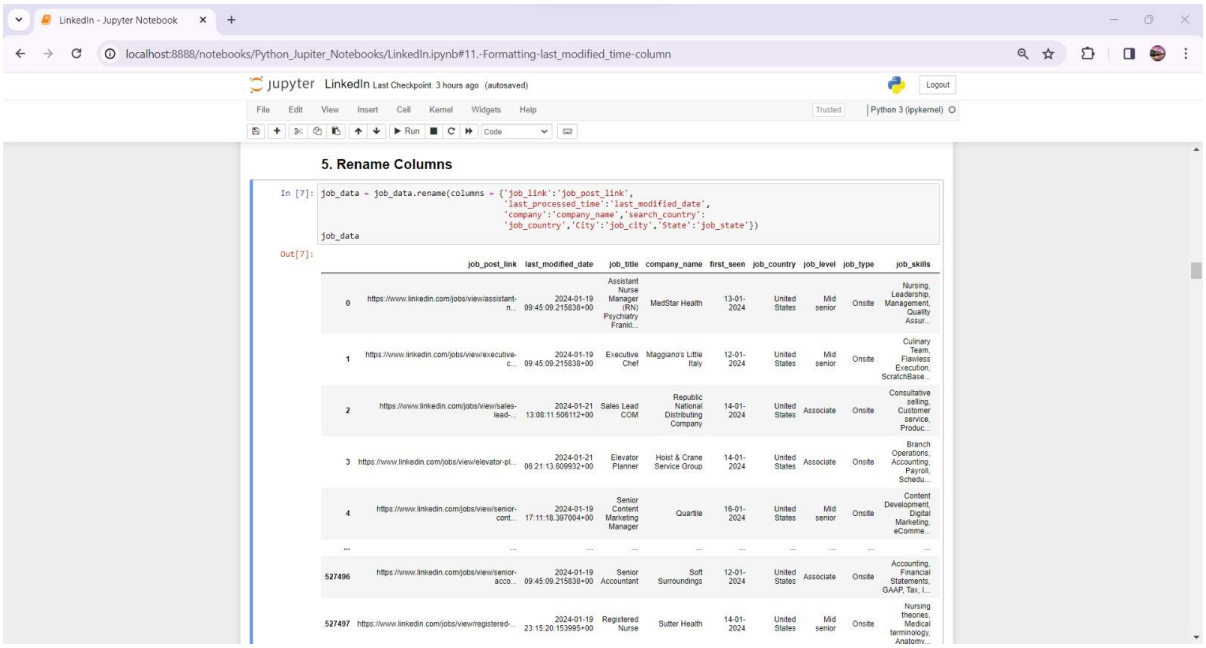
Out[6]:

	job_link	last_processed_time	job_title	company	search_country	job_level	job_type	job_skills	City
0	https://www.linkedin.com/jobs/view/assistant-n...	2024-01-19 09:45:09.215838+00	Assistant Nurse Manager (RN) Psychiatry Frankl...	MedStar Health	United States	Mid senior	Onsite	Nursing, Leadership, Management, Quality Assur...	Baltimore
1	https://www.linkedin.com/jobs/view/executive-c...	2024-01-19 09:45:09.215838+00	Executive Chef	Maggiano's Little Italy	United States	Mid senior	Onsite	Culinary Team, Flawless Execution, ScratchBase...	King of Prussia
2	https://www.linkedin.com/jobs/view/sales-lead-...	2024-01-21 13:08:11.506112+00	Sales Lead COM	Republic National Distributing Company	United States	Associate	Onsite	Consultative selling, Customer service, Produc...	Grand Prairie
3	https://www.linkedin.com/jobs/view/elevator-pl...	2024-01-21 06:21:13.609932+00	Elevator Planner	Hoist & Crane Service Group	United States	Associate	Onsite	Branch Operations, Accounting, Payroll, Schedu...	Greater New Orleans Region
4	https://www.linkedin.com/jobs/view/senior-cont...	2024-01-19 17:11:18.397004+00	Senior Content Marketing Manager	Quartile	United States	Mid senior	Onsite	Content Development, Digital Marketing, eComme...	New York
...	...	...	...	...	...	...	...	...	...
527496	https://www.linkedin.com/jobs/view/senior-acco...	2024-01-19 09:45:09.215838+00	Senior Accountant	Soft Surroundings	United States	Associate	Onsite	Accounting, Financial Statements, GAAP, Tax, I...	Greater St. Louis
527497	https://www.linkedin.com/jobs/view/registered-...	2024-01-19 23:15:20.153995+00	Registered Nurse	Sutter Health	United States	Mid senior	Onsite	Nursing theories, Medical terminology	Ewa Beach

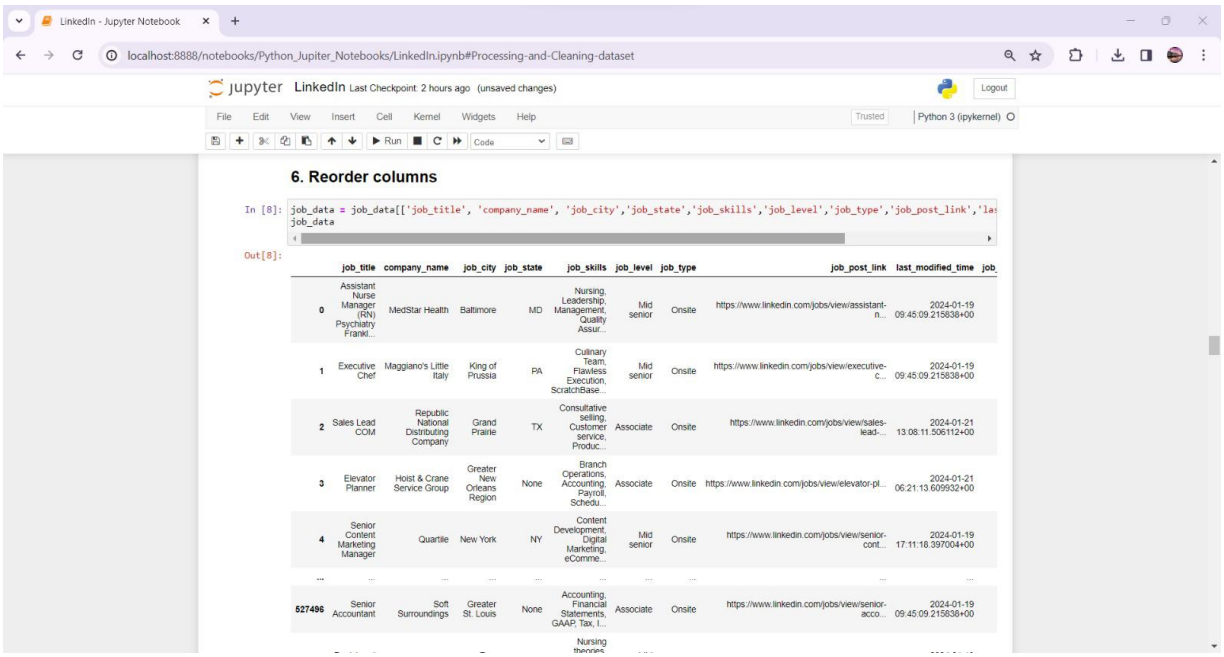
3.5 RENAME COLUMNS

To better understand the relationship between the data and the feature we are renaming the columns below.

- 'job_link' -> 'job_post_link'
- 'last_processed_time' -> 'last_modified_date'
- 'company' -> 'company_name'
- 'search_country' -> 'job_country'
- 'City' -> 'job_city'
- 'State' -> 'job_state'

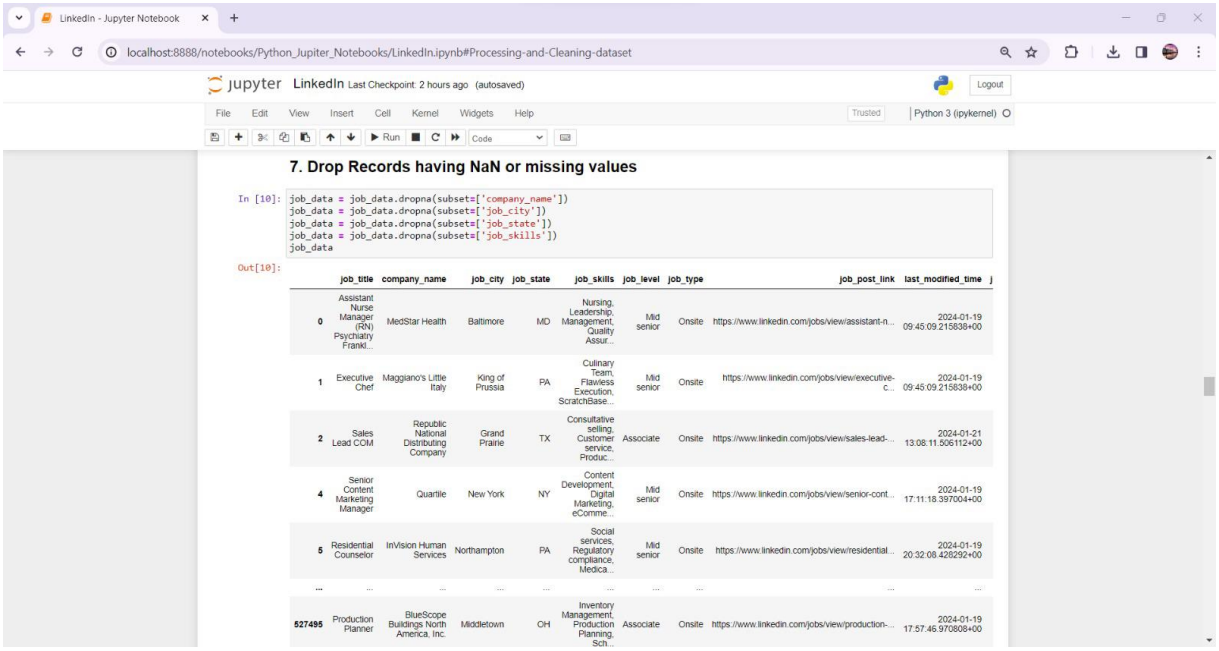


3.6 REORDER COLUMNS

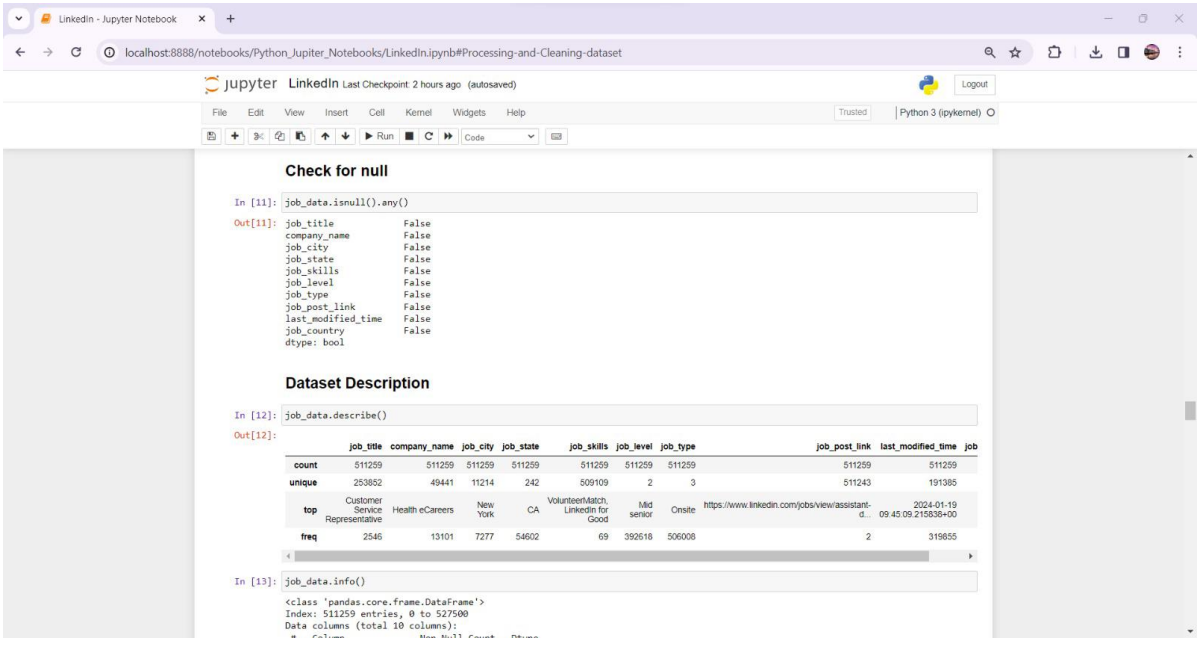


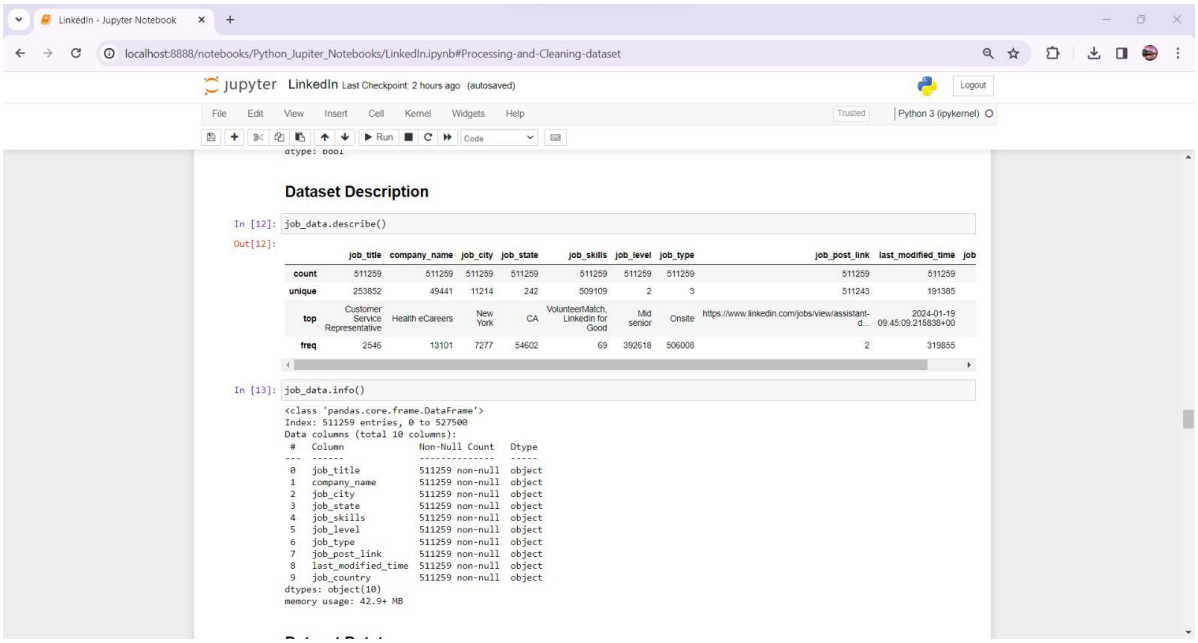
3.7 DROP RECORDS HAVING ‘NaN’ OR MISSING VALUES

Removing or replacing Null values avoids the risk of bias due to missing values.



All rows with NaN values are removed.

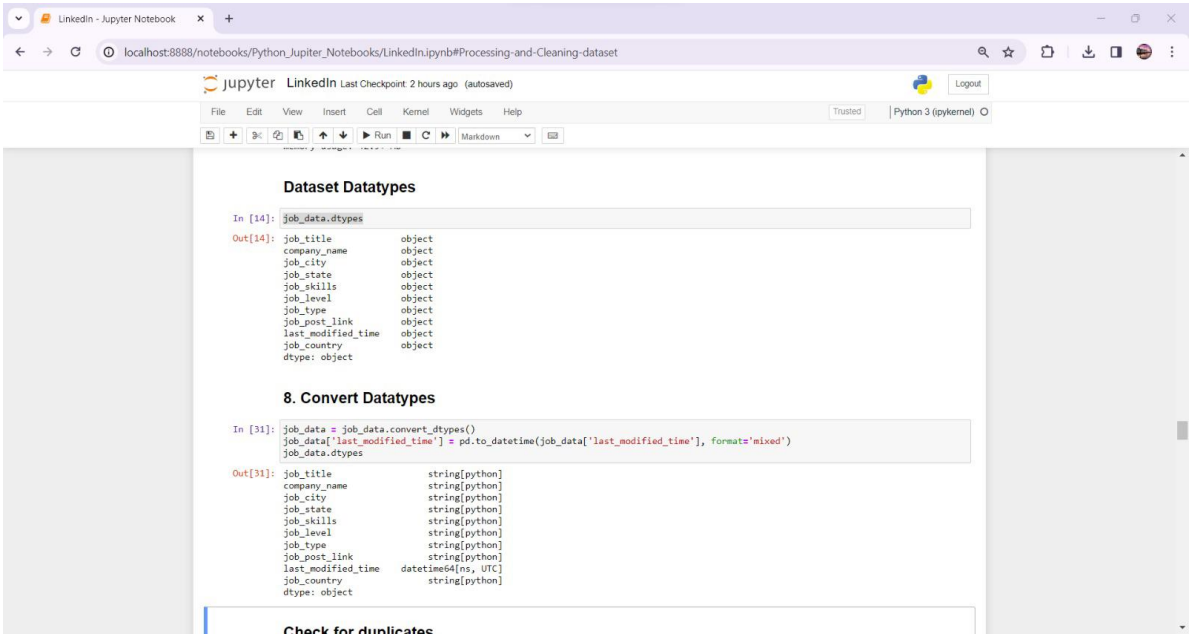




3.8 CONVERTING DATATYPES

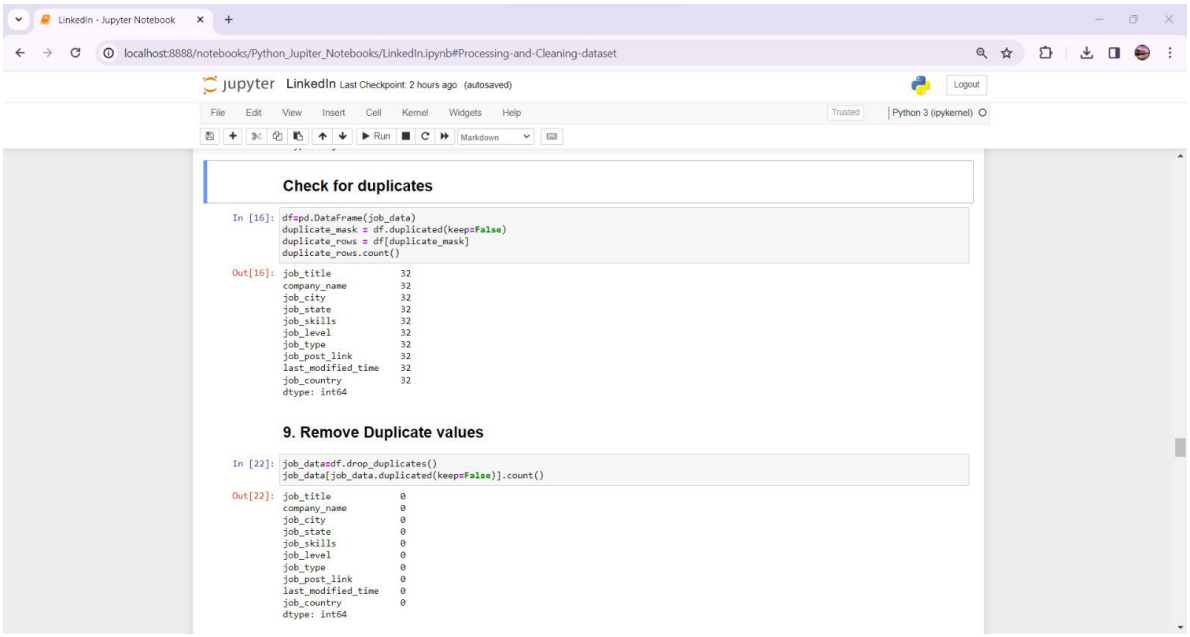
Prevents errors that may arise from using incompatible data types during calculations or operations.

Datatype of ‘last_modified_time’ is changed to ‘datetime’



3.9 REMOVE DUPLICATE VALUES

- Eliminating identical entries that appear multiple times in the dataset, ensuring data accuracy and avoiding skewing results.
- ‘duplicate_rows’ displays the count of the number of duplicate present in the dataset.
- The duplicate records are removed using the built-in method(drop_duplicate).

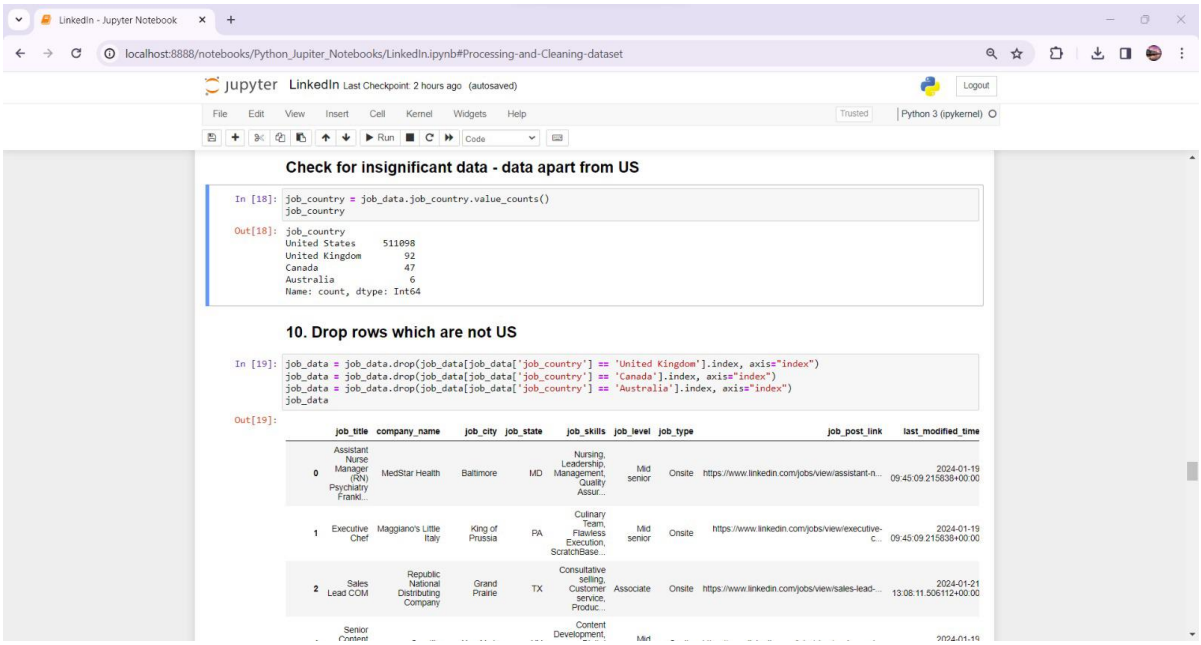


3.10 DROP ROWS WHICH ARE NOT ‘United States’

When we group and see the number of rows for each country. ‘United States’ has huge number of records and the other countries numbers are very less compared to it. Hence, the other countries data is dropped to have good correlation between data.

We are using data only based on country=’USA’.

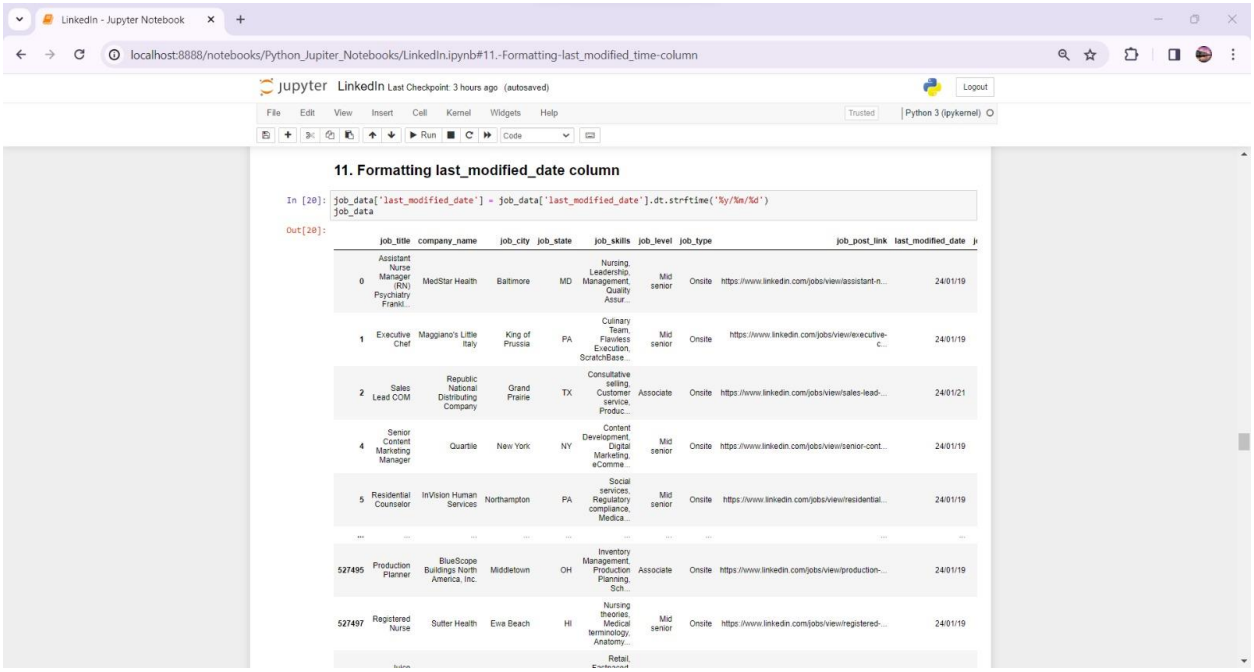
The dataset has records from countries other than our required nation. We are dropping these records for better training and accuracy of the models.



3.11 FORMATTING ‘last_modified_date’ COLUMN

The records for the column ‘last_modified_date’ is of the format YYYY/MM/DD Time.

Time is irrelevant, and we are changing the format to YYYY/MM/DD.



AFTER CLEANING:

No. of Rows	511, 098
No. of Columns	11

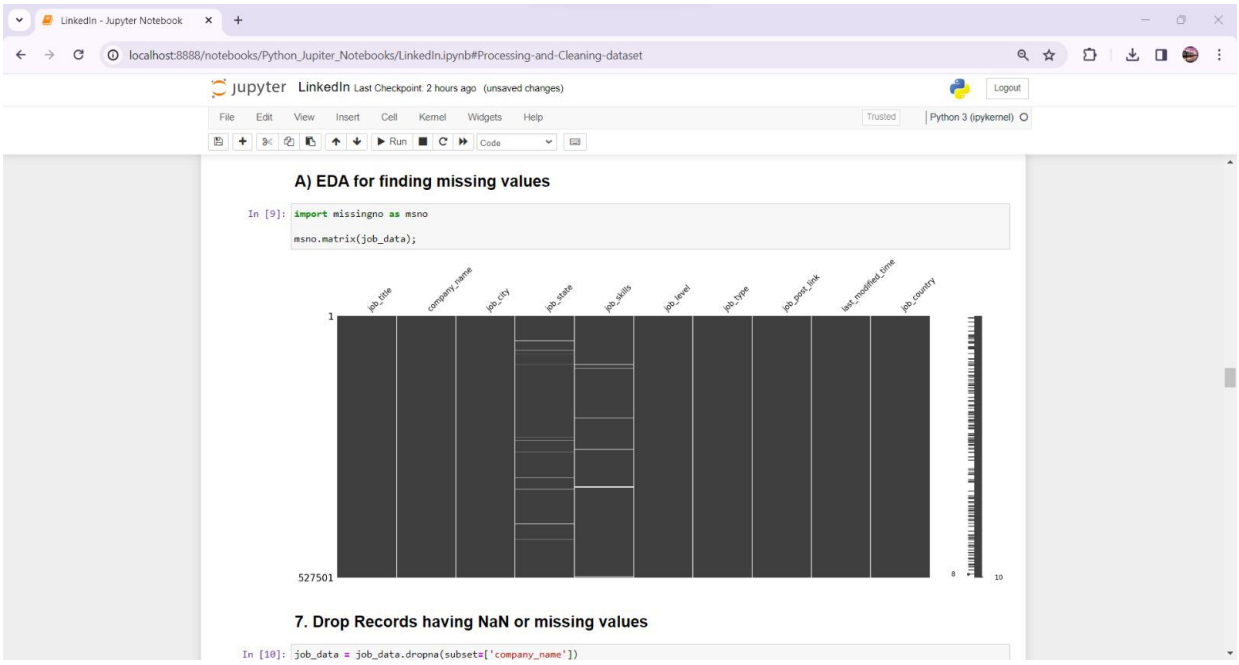
4. EXPLORATORY DATA ANALYSIS:

4.1 EDA FOR FINDING MISSING VALUES

A graph is plotted to see the missing values present in each column. The lines present for columns in the graph tell the spread of missing values across rows for each column.

As we see in the 'job_state' and 'job_skills' there are few missing values for some records.

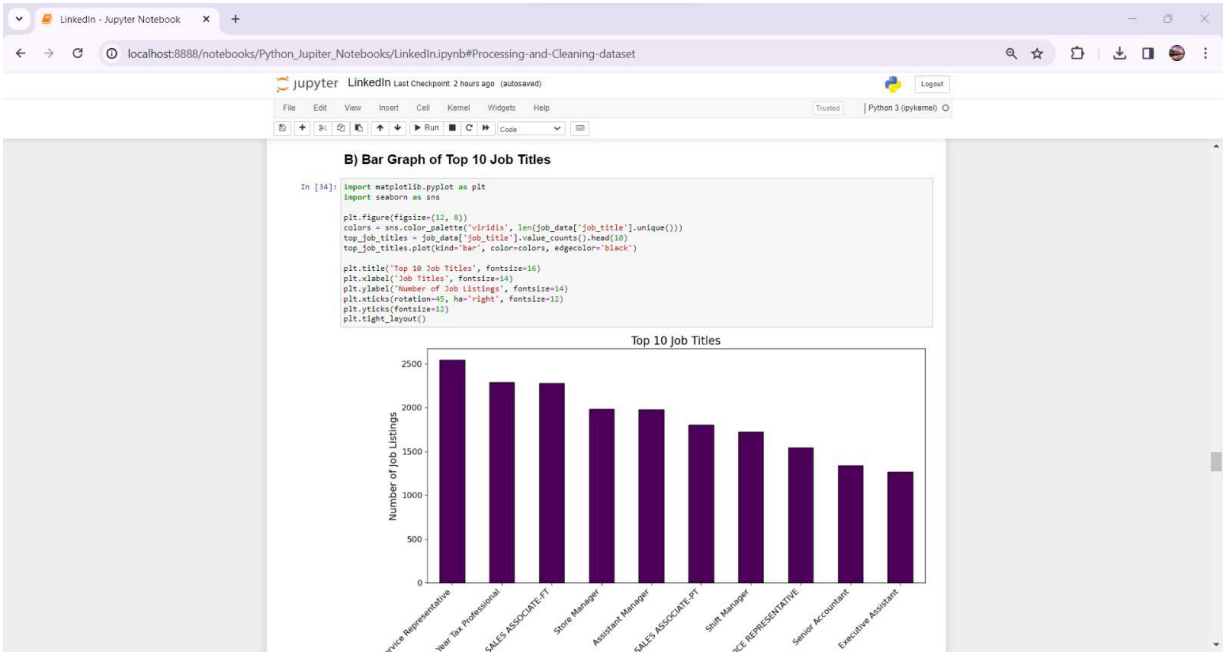
Since, both are unique and play a vital role for prediction. In the data cleaning process, we have removed the missing values rows since they will be less significant.



4.2 TOP 10 JOB TITLES – BAR GRAPH

From the below Bar Graph, we can infer the ‘Top 10 Job Titles’ which are posted and the frequency of the same.

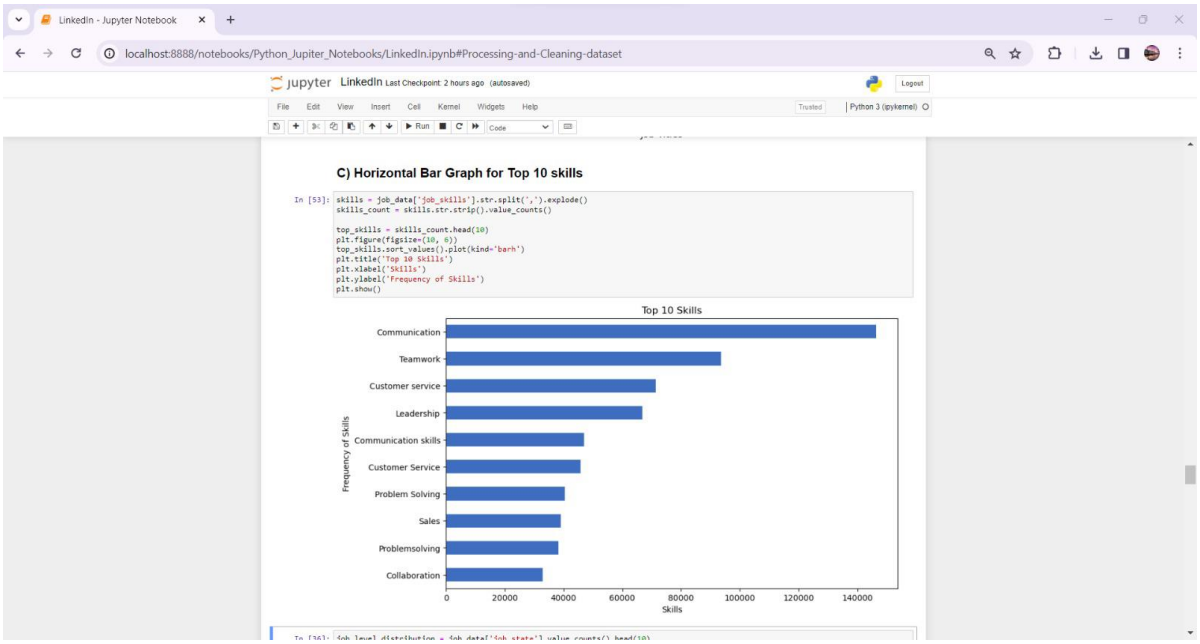
Among these top titles we have three management positions “Store Manager”, “Assistant Manager” and “Shift Manager”. Also, two “Lead Associate” positions are there.



4.3 TOP 10 SKILLS – HORIZONTAL BAR GRAPH

The ‘Top 10 Skills’ which are mentioned as required and their frequency is represented in the Horizontal Bar graph.

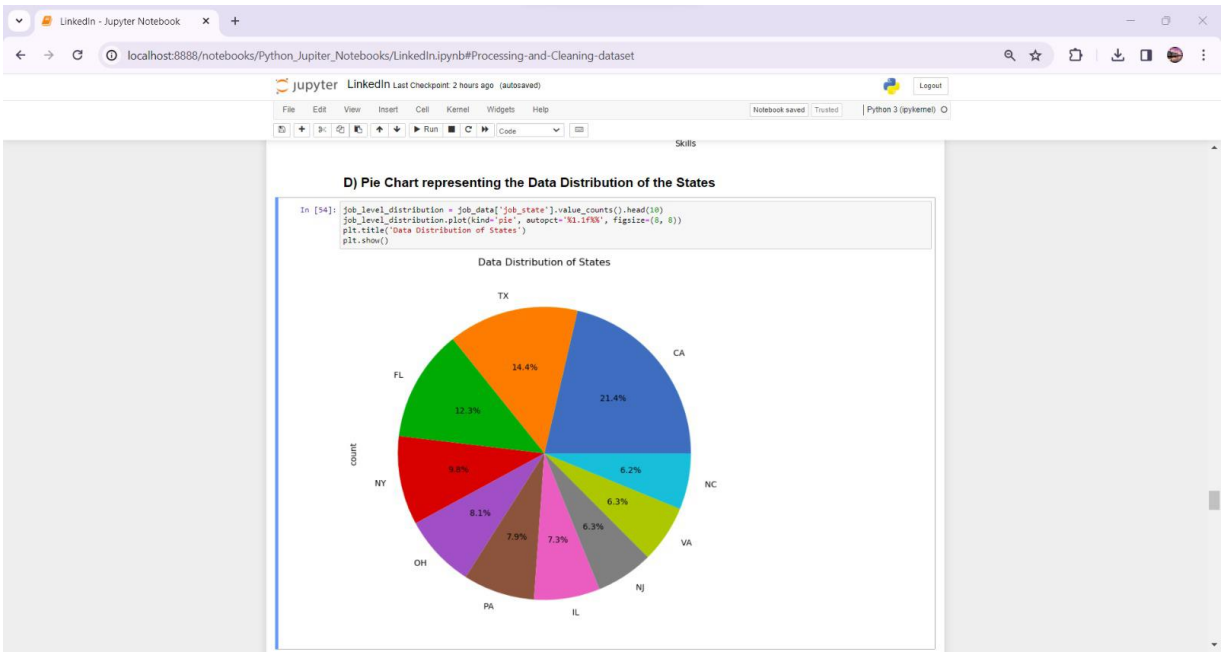
We can infer that, Basic skills like “Communication”, “Teamwork” and “Customer service” which are essential are mentioned for most of the companies.



4.4 DATA DISTRIBUTION AMONG STATES – PIE CHART

The Pie chart represents the percentage of data of the Whole dataset for each State.

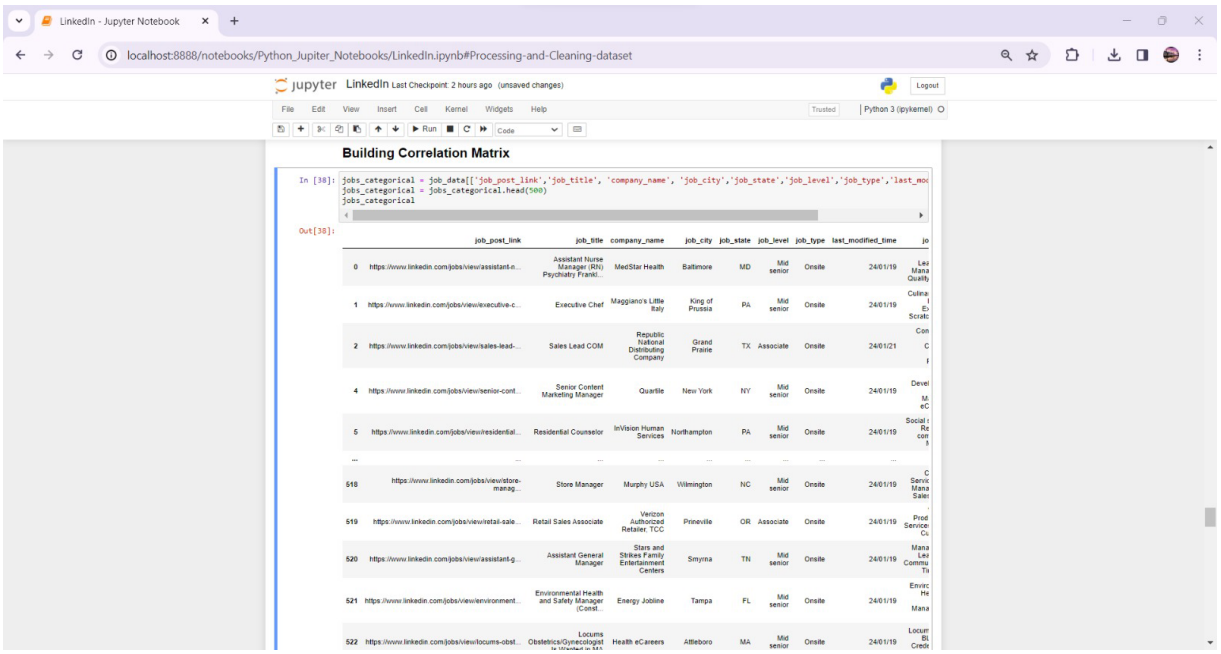
Top four states which have more job openings are “Texas”, “California”, “Florida” and “New York”.

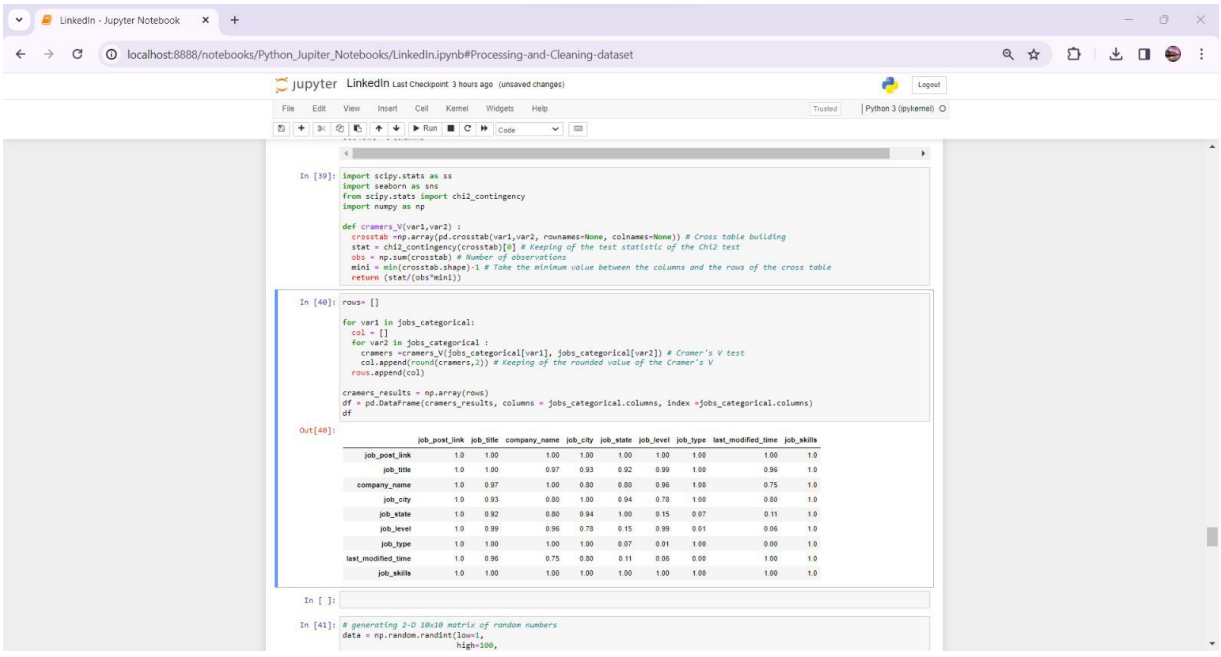


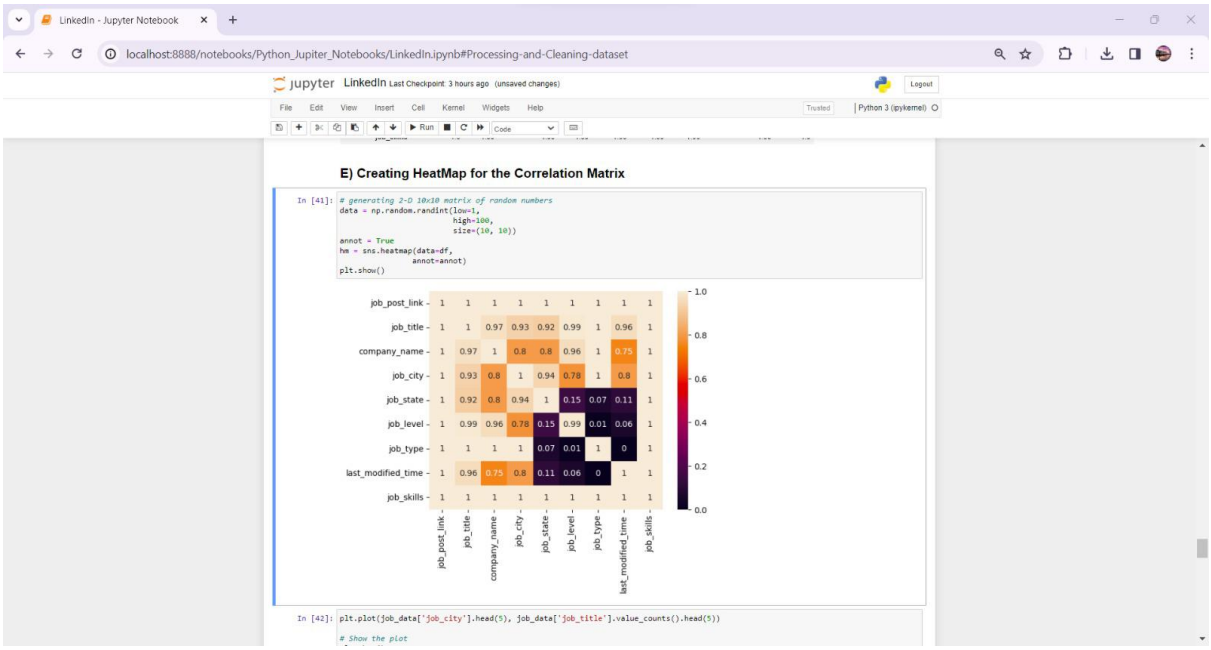
4.5 CORRELATION MATRIX – HEATMAP

The correlation matrix is calculated with Cramer's V and a chi-square test is also performed.

With the correlation matrix the statistically significant relations between the rows are identified and plotted as a heatmap.


LinkedIn - Jupyter Notebook
localhost:8888/notebooks/Python_Jupiter_Notebooks/LinkedIn.ipynb#Processing-and-Cleaning-dataset
jupyter LinkedIn Last Checkpoint: 2 hours ago (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3 (ipykernel)
Run
Code
Building Correlation Matrix
In [38]:
jobs_categorical = job_data[['job_post_link','job_title', 'company_name', 'job_city','job_state','job_level','job_type','last_mo
jobs_categorical = jobs_categorical.head(500)
jobs_categorical
Out[38]:
job_post_link job_title company_name job_city job_state job_level job_type last_modified_time jo
0 https://www.linkedin.com/jobs/view/assistant-n... Assistant Nurse Manager (RN) Psychiatry Franki... MedStar Health Baltimore MD Mid senior Onsite 24/01/19 Lea Mana Quality
1 https://www.linkedin.com/jobs/view/executive-c... Executive Chef Maggiano's Little Italy King of Prussia PA Mid senior Onsite 24/01/19 Culina I Ex Scratc
2 https://www.linkedin.com/jobs/view/sales-lead-... Sales Lead COM Republic National Distributing Company Grand Prairie TX Associate Onsite 24/01/21 Con C F
4 https://www.linkedin.com/jobs/view/senior-cont... Senior Content Marketing Manager Quartile New York NY Mid senior Onsite 24/01/19 Devel M eC
5 https://www.linkedin.com/jobs/view/residential... Residential Counselor InVision Human Services Northampton PA Mid senior Onsite 24/01/19 Social Re con h
518 https://www.linkedin.com/jobs/view/store-manag... Store Manager Murphy USA Wilmington NC Mid senior Onsite 24/01/19 C Servic Mana Sales
519 https://www.linkedin.com/jobs/view/retail-sale... Retail Sales Associate Verizon Authorized Retailer, TCC Prineville OR Associate Onsite 24/01/19 Prod Service Cu
520 https://www.linkedin.com/jobs/view/assistant-g... Assistant General Manager Stars and Strikes Family Entertainment Centers Smyrna TN Mid senior Onsite 24/01/19 Mana Lea Commu Th
521 https://www.linkedin.com/jobs/view/environment... Environmental Health and Safety Manager (Const... Energy Jobline Tampa FL Mid senior Onsite 24/01/19 Enviro He Mana
522 https://www.linkedin.com/jobs/view/locums-obst... Locums Obstetrics/Gynecologist Health eCareers Attleboro MA Mid senior Onsite 24/01/19 Locum BL Credit

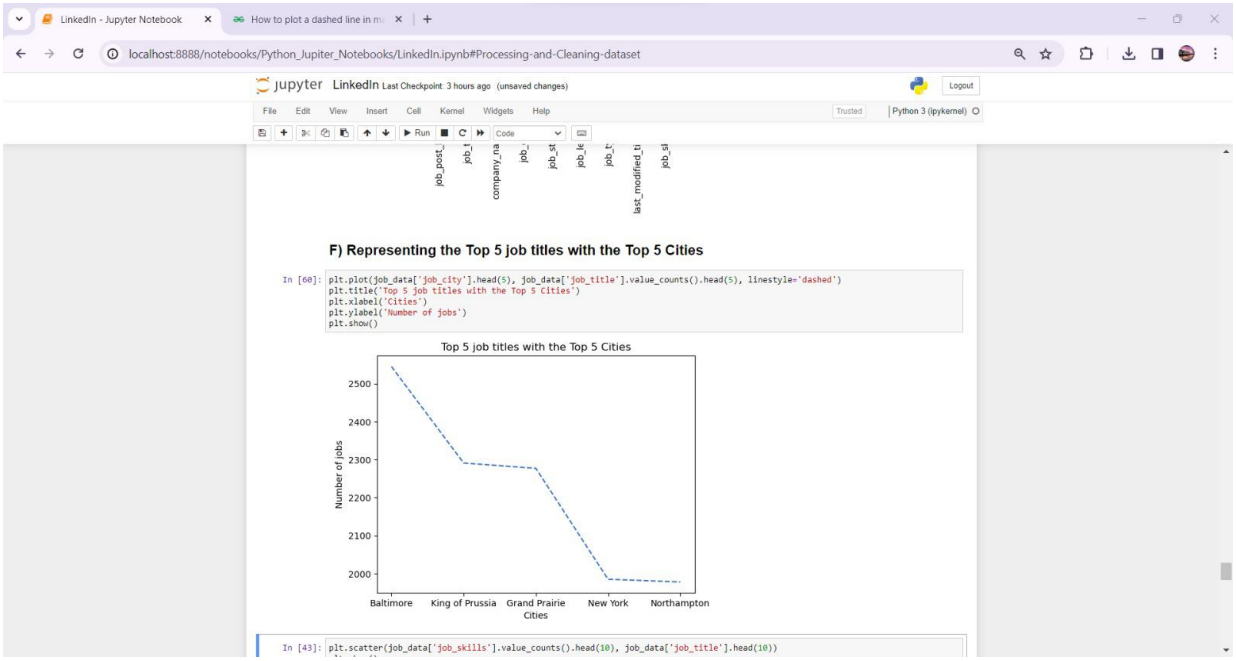

LinkedIn - Jupyter Notebook
localhost:8888/notebooks/Python_Jupiter_Notebooks/LinkedIn.ipynb#Processing-and-Cleaning-dataset
jupyter LinkedIn Last Checkpoint: 3 hours ago (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3 (ipykernel)
Run
Code
In [39]:
import scipy.stats as ss
import seaborn as sns
from scipy.stats import chi2_contingency
import numpy as np
def cramers_V(var1,var2) :
crosstab =np.array(pd.crosstab(var1,var2, rownames=None, colnames=None)) # Cross table building
stat = chi2_contingency(crosstab)[0] # Keeping of the test statistic of the Chi2 test
obs = np.sum(crosstab) # Number of observations
mini = min(crosstab.shape)-1 # Take the minimum value between the columns and the rows of the cross table
return (stat/(obs*mini))
In [40]:
rows= []
for var1 in jobs_categorical:
col = []
for var2 in jobs_categorical :
cramers =cramers_V(jobs_categorical[var1], jobs_categorical[var2]) # Cramer's V test
col.append(round(cramers,2)) # Keeping of the rounded value of the Cramer's V
rows.append(col)
cramers_results = np.array(rows)
df = pd.DataFrame(cramers_results, columns = jobs_categorical.columns, index =jobs_categorical.columns)
df
Out[40]:
job_post_link job_title company_name job_city job_state job_level job_type last_modified_time job_skills
job_post_link 1.0 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.0
job_title 1.0 1.00 0.97 0.93 0.92 0.99 1.00 0.96 1.0
company_name 1.0 0.97 1.00 0.80 0.80 0.96 1.00 0.75 1.0
job_city 1.0 0.93 0.80 1.00 0.94 0.78 1.00 0.80 1.0
job_state 1.0 0.92 0.80 0.94 1.00 0.15 0.07 0.11 1.0
job_level 1.0 0.99 0.96 0.78 0.15 0.99 0.01 0.06 1.0
job_type 1.0 1.00 1.00 1.00 0.07 0.01 1.00 0.00 1.0
last_modified_time 1.0 0.96 0.75 0.80 0.11 0.06 0.00 1.00 1.0
job_skills 1.0 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.0
In [ ]:
In [41]:
# generating 2-D 10x10 matrix of random numbers
data = np.random.randint(low=1,
high=100,



4.6 NO OF JOB TITLES VS TOP 5 CITIES – LINE GRAPH

Below graph represents the graph of No of different Job titles in Top 5 Cities.

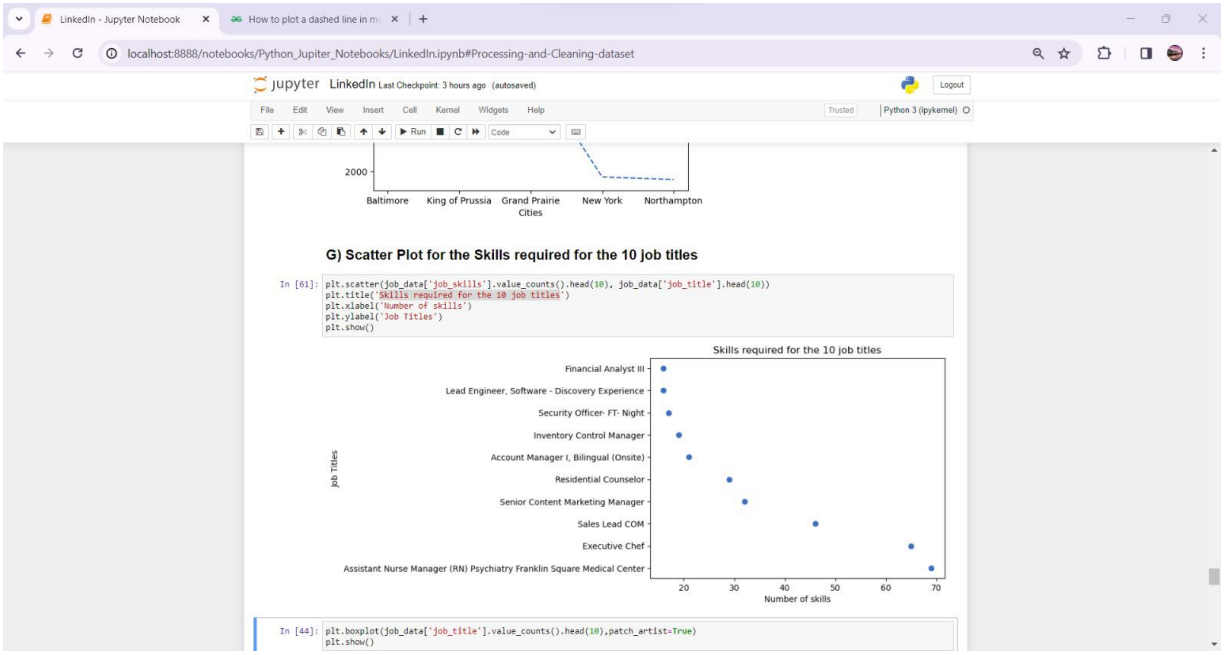
"Baltimore" city has the most job openings in different domains. "Northampton" has comparatively less diverse job openings.



4.7 TOP 10 JOB TITLES VS NUMBER OF SKILLS – SCATTER PLOT

Below graph depicts the number of skills required for Top 10 Job titles.

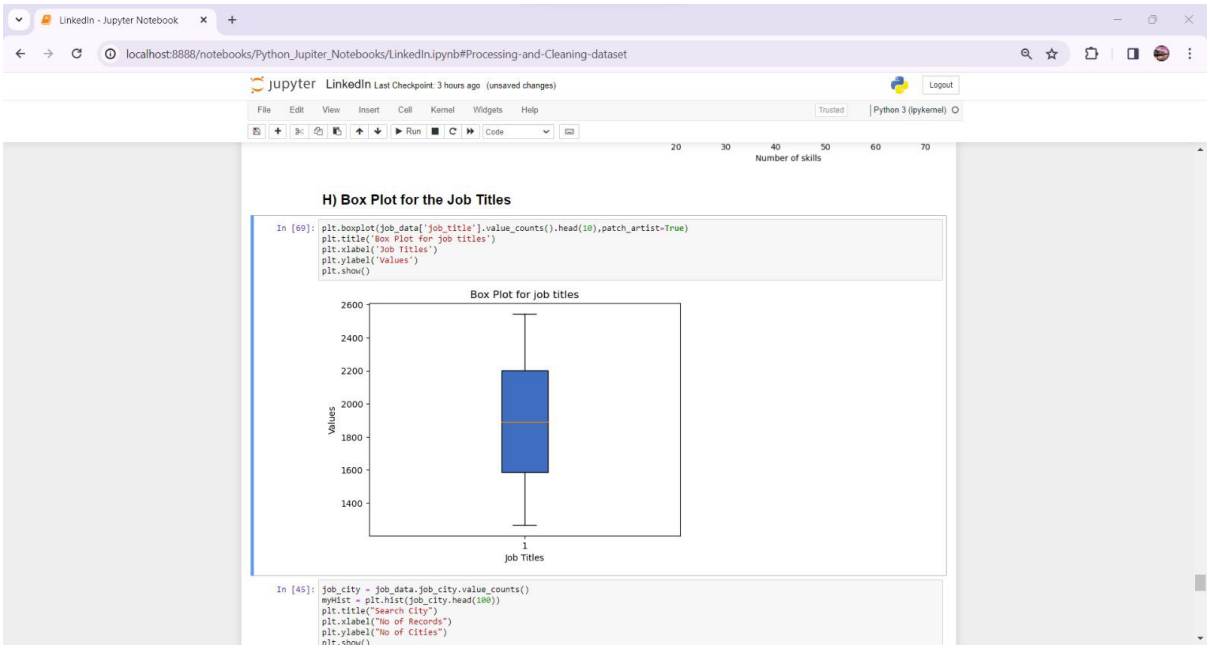
The graph shows that “Executive Chef”, and “Assistant Nurse Manager” require more skills.



4.8 JOB TITLES - BOX PLOT

Box plots are commonly used to identify outliers, visualize data dispersion, compare distributions, and detect skewness or asymmetry in the data distribution.

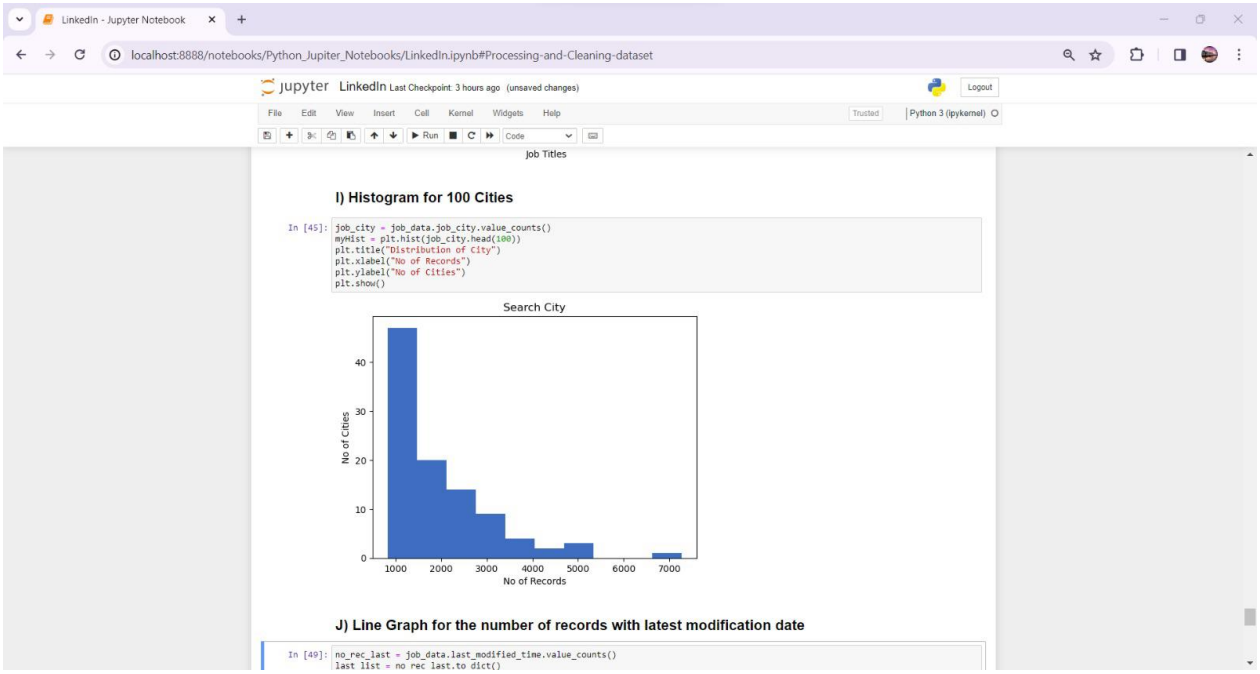
It consists of a box that represents the interquartile range (IQR) of the data, with a line inside indicating the median.



4.9 HISTOGRAM FOR 100 CITIES

The Histogram represents plots No of Records across the No od Cities having them.

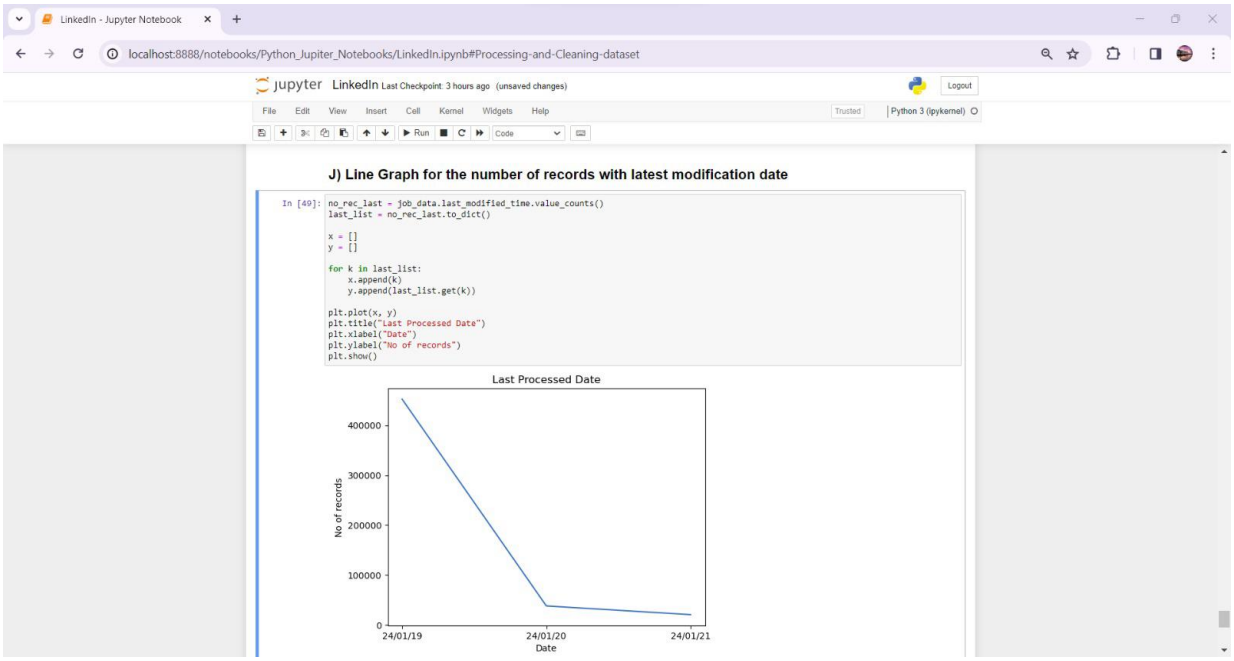
We can infer that more than 40 cities have more than 1000 job openings.

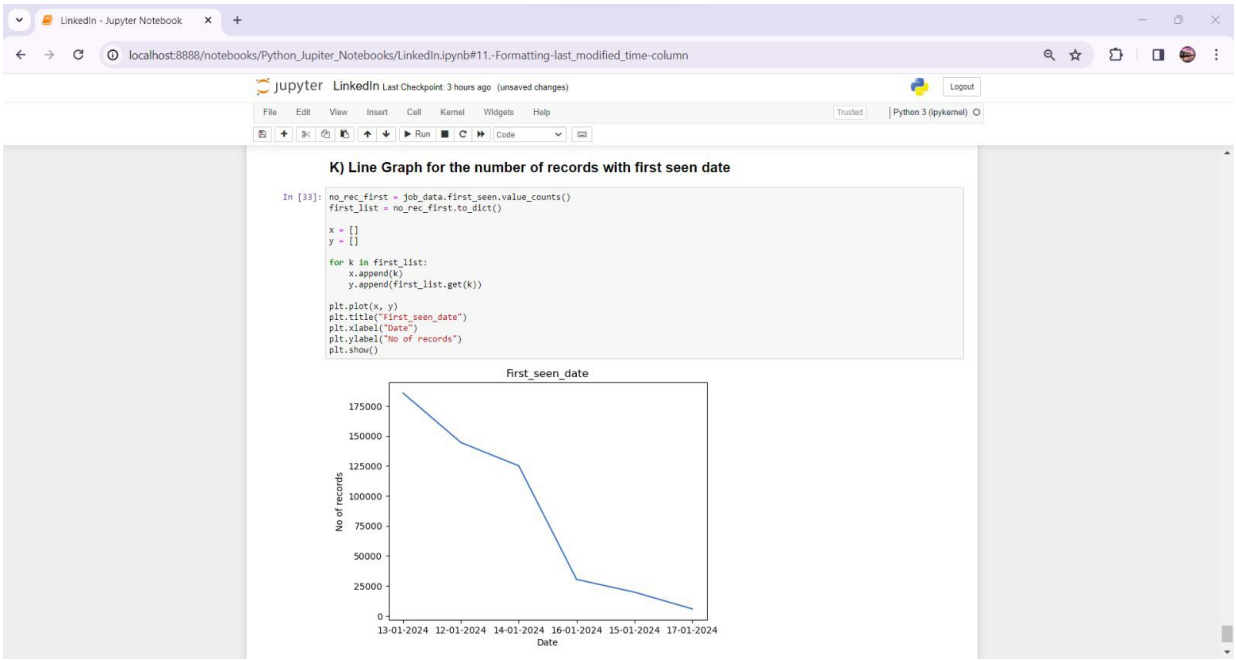


4.10 DATA COLLECTION PERIOD

Two graphs are plotted to infer when the data was modified last and which period the data has been collected.

The data is collected between 13th to 17th January 2024. The last modified date of most of the data lies between 19th to 21st January 2024.





Conclusion:

In conclusion, our Personalized Job Recommendation project has undergone meticulous data cleaning and exploratory data analysis (EDA), laying a robust foundation for the development of an effective recommendation system. Through eleven comprehensive data cleaning steps, we have ensured the accuracy, consistency, and reliability of our dataset. These steps encompassed handling missing values, removing duplicates, standardizing data formats, correcting errors, addressing outliers, and ensuring data integrity, among others.

Furthermore, our exploration of the dataset involved ten distinct EDA techniques, each providing unique insights into the underlying patterns and characteristics of the data. Leveraging a diverse range of graphical representations including box plots, bar graphs, pie charts, histograms, scatterplots, and heat maps, we gained a comprehensive understanding of various aspects such as distribution, relationships, trends, and anomalies within the dataset. These visualizations have enabled us to identify key features, understand data distributions, detect outliers, and uncover potential correlations, laying the groundwork for informed decision-making in the subsequent stages of our project.

Project Phase #2

Team Member 1: DHIVYASHREE SIVAPRAKASAM (UBIT – dsivapra, Person number - 50560688)

Team Member 2: SHALINI ANANTHAVEL JAYALAKSHMI (UBIT – ananthav, Person number - 50560497)

Team Member 3: GANESH PRABAKARAN (UBIT – ganeshpr, Person number - 50560751)

Problem statement:

The abundance of postings often overwhelms job seekers, hindering their ability to find relevant opportunities. Hence, there's a critical need for an intelligent system to analyze user data and offer personalized job recommendations. **This project aims to address information overload, personalize recommendations, and enhance job search efficiency by creating a system that utilizes machine learning and user data to deliver tailored job suggestions.**

Input:

- The user provides a set of skills to get the customized job titles and its LinkedIn url along with other details of the post
- Dataset Source:
- Kaggle : https://www.kaggle.com/datasets/asaniczka/1-3m-linkedin-jobs-and-skills-2024
- The data cleaning and EDA part is done in phase 1 which is the input for performing the clustering and implementing the algorithms.

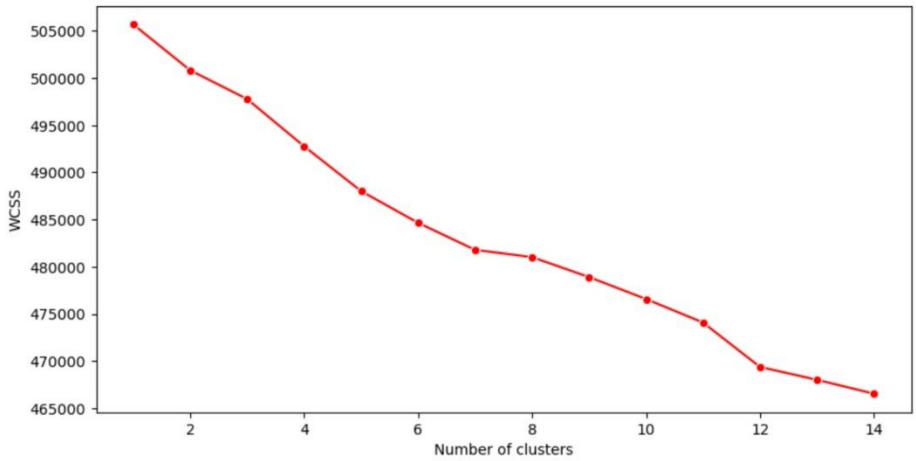
Expected output:

The job cluster is identified from the input skills from the user and displays the job posting details with the relevant job titles that match the skills with highest accuracy.

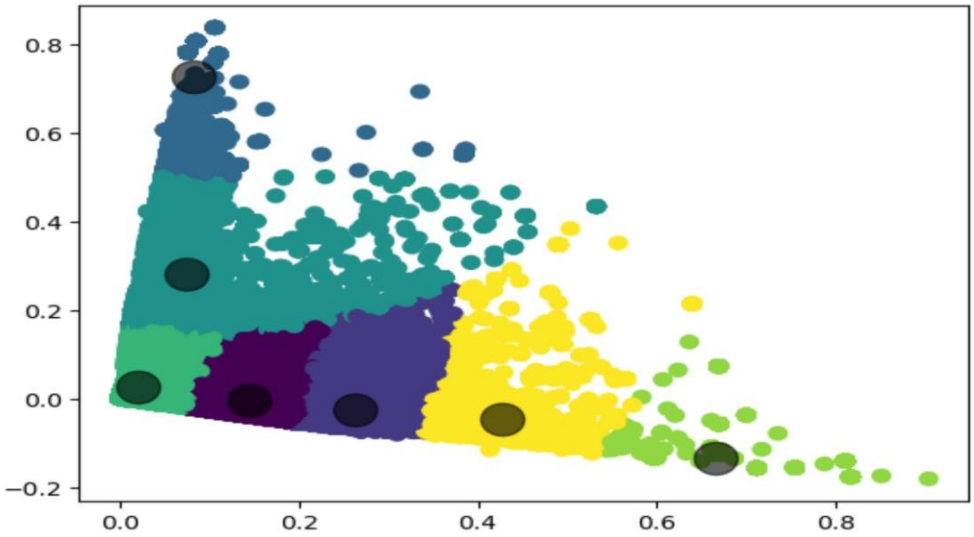
CLUSTERING:

- The TF-IDF vectorization converts job titles into numerical representations and iterates through different numbers of clusters using K-means clustering to find the optimal number of clusters by minimizing the within-cluster sum of squares (WCSS).

- The WCSS measures the compactness of clusters, with lower values indicating tighter clusters.
- By plotting the number of clusters against the WCSS, the code visually identifies the **"elbow point"(our case its k =7)** indicating the optimal number of clusters where further partitioning yields diminishing returns in terms of reducing WCSS.



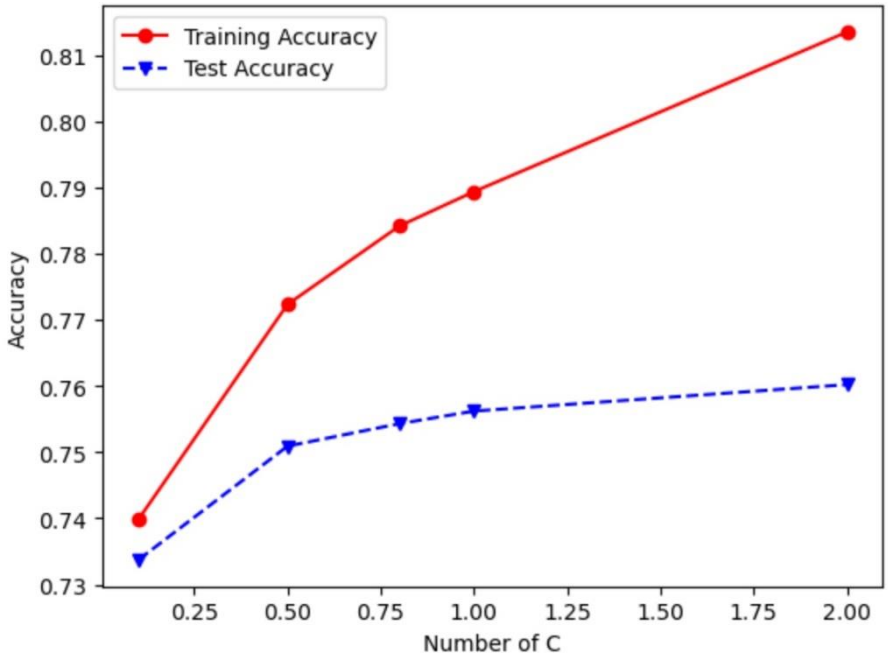
- After performing dimensionality reduction using TruncatedSVD and the clustering is done using KMeans clustering algorithm. Since the k value is 7, the data is clustered into 7 clusters and are labeled as "Technical Professional", "Restaurant Management", "Sales Lead or Customer Service Associate", "Business Administration", "Healthcare Worker", "Customer Service Specialist", "Executive Dean Healthcare".



- The Label column is added where for each record we can see the labels mentioned that is generated using K-means algorithm.

ALGORITHMS:

1) **LOGISTIC REGRESSION**
   a) **Why this algorithm?**
      - Logistic regression's coefficients reveal the impact of each skill on the probability of a good fit.
      - In the initial stages, labeled data for complex recommendation algorithms might be limited. Logistic regression can effectively utilize datasets with labeled information on user-job suitability. Due to its simplicity, logistic regression can be trained and deployed quickly.
   b) **Tuning and Training the model**
      - The logistic regression model is tuned by iterating over a range of regularization parameter values (Cparamrange).
      - For each value of C, a logistic regression model is trained on the training data (X_train and y_train). The model's accuracy is then evaluated on both the training and testing datasets



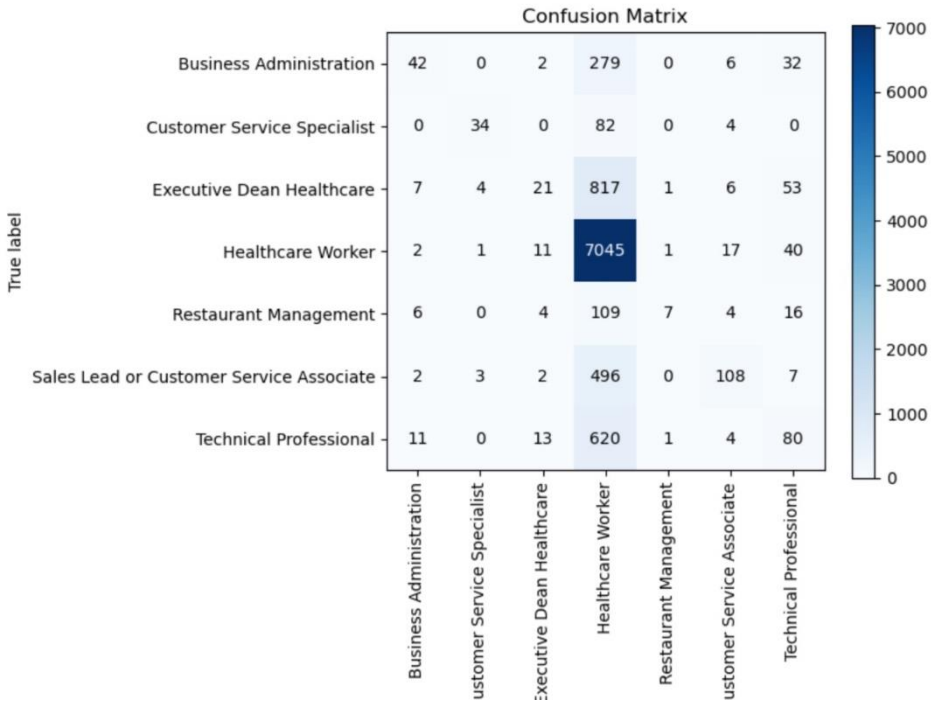
   c) **Effectiveness of Algorithm**
      - The model achieved acceptable accuracy, precision, and recall scores, indicating its ability to identify some good-fit job opportunities based on user skills.

Accuracy of Logistic Regression: 0.7337

	precision	recall	f1-score	support
Business Administration	0.60	0.12	0.19	361
Customer Service Specialist	0.81	0.28	0.42	120
Executive Dean Healthcare	0.40	0.02	0.04	909
Healthcare Worker	0.75	0.99	0.85	7117
Restaurant Management	0.70	0.05	0.09	146
Sales Lead or Customer Service Associate	0.72	0.17	0.28	618
Technical Professional	0.35	0.11	0.17	729
accuracy			0.73	10000
macro avg	0.62	0.25	0.29	10000
weighted avg	0.68	0.73	0.65	10000

d) Visualizaion

Out[16]: <Axes: title={'center': 'Confusion Matrix'}, xlabel='Predicted label', ylabel='True label'>



2) DECISION TREE

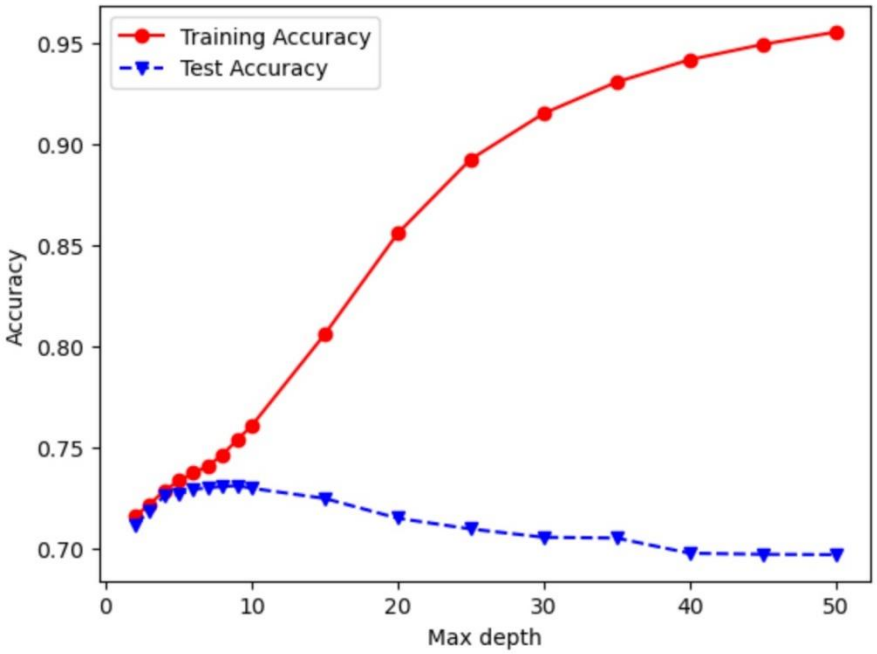
a) Why this algorithm?

- Decision trees can handle multi-class classification, allowing them to recommend jobs with varying degrees of suitability (e.g., excellent match, good fit, consider for further exploration). This is crucial for providing a more nuanced recommendation experience.

b) Tuning and Training the model

- To tune and train a Decision Tree classifier model, a range of maximum depths (maxdepths) is defined.

- This range determines the complexity of the decision tree. Each depth value is iterated over, creating a DecisionTreeClassifier object with the specified depth.
- This classifier is then trained on the provided training data (X_train and y_train).



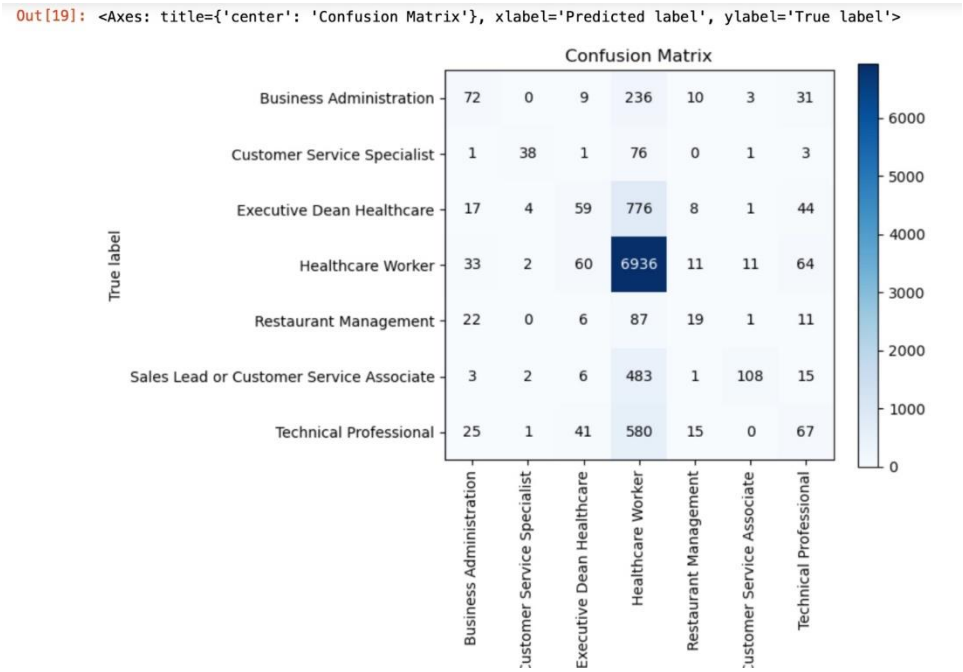
c) **Effectiveness of Algorithm**

- By handling multi-class classification and offering a more nuanced representation of job suitability, decision trees can potentially lead to more accurate recommendations

Accuracy of Decision tree: 0.7299

	precision	recall	f1-score	support
Business Administration	0.42	0.20	0.27	361
Customer Service Specialist	0.81	0.32	0.46	120
Executive Dean Healthcare	0.32	0.06	0.11	909
Healthcare Worker	0.76	0.97	0.85	7117
Restaurant Management	0.30	0.13	0.18	146
Sales Lead or Customer Service Associate	0.86	0.17	0.29	618
Technical Professional	0.29	0.09	0.14	729
accuracy			0.73	10000
macro avg	0.54	0.28	0.33	10000
weighted avg	0.67	0.73	0.66	10000

d) Visualization



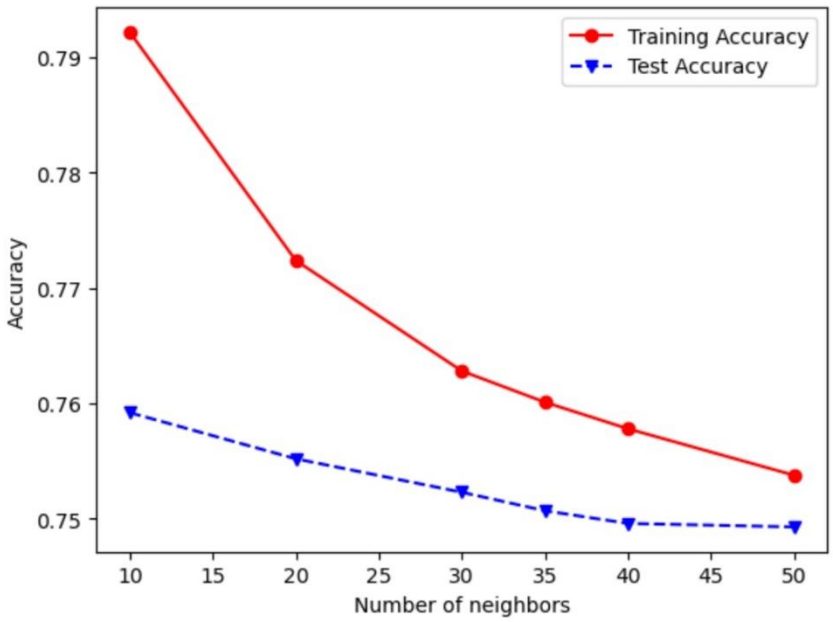
3) K-NEAREST NEIGHBOR

a) Why this algorithm?

- KNN leverages the concept of collaborative filtering. It recommends jobs based on the preferences (skills) of similar users.

b) Tuning and Training the model

- To tune and train the K-Nearest Neighbors (KNN) classifier model, a range of numNeighbors is defined to represent different numbers of neighbors for classification.
- Each value of numNeighbors is iterated over, creating a KNeighborsClassifier object with the specified number of neighbors and using the Minkowski distance metric with p=2 (Euclidean distance).
- The classifier is then trained on the provided training data (X_train and y_train).



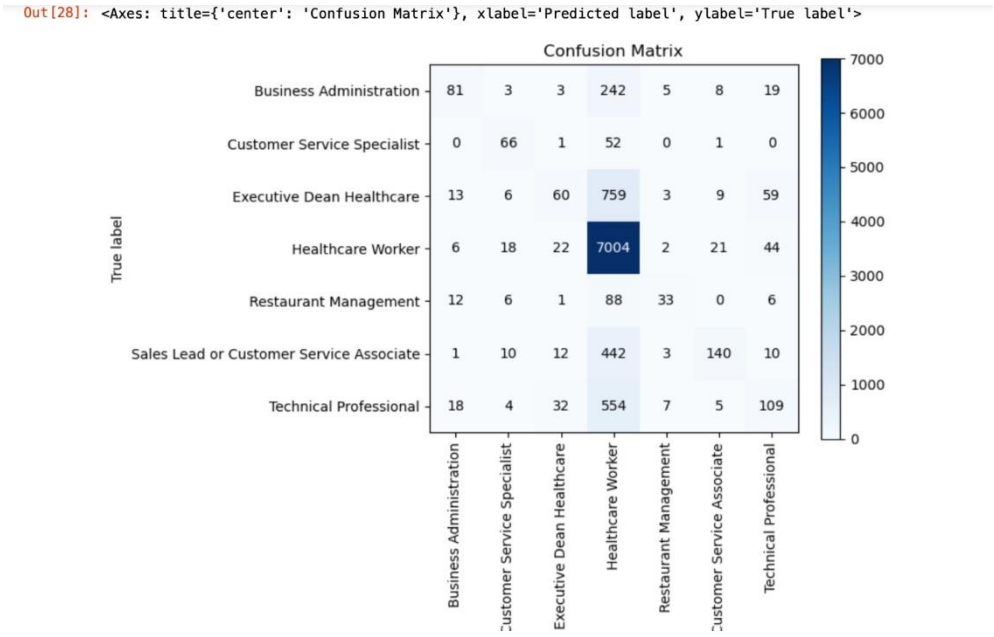
c) **Effectiveness of Algorithm**

- KNN can handle both numerical and categorical features naturally. This versatility is advantageous when dealing with diverse types of user data, such as skills, experience, location preferences, and job categories.
- Localized Recommendations: KNN provides localized recommendations by considering only the nearest neighbors of a given user. This can be beneficial for recommending jobs based on specific user preferences or contexts, such as location, industry, or job type.

Accuracy of KNN: 0.7493

	precision	recall	f1-score	support
Business Administration	0.62	0.22	0.33	361
Customer Service Specialist	0.58	0.55	0.57	120
Executive Dean Healthcare	0.46	0.07	0.12	909
Healthcare Worker	0.77	0.98	0.86	7117
Restaurant Management	0.62	0.23	0.33	146
Sales Lead or Customer Service Associate	0.76	0.23	0.35	618
Technical Professional	0.44	0.15	0.22	729
accuracy			0.75	10000
macro avg	0.61	0.35	0.40	10000
weighted avg	0.70	0.75	0.69	10000

d) **Visualization**



4) NAIVE BAYES

a) Why this algorithm?

- Naive Bayes is a relatively simple algorithm with easy interpretability.

b) Tuning and Training the model

- To train the Gaussian Naive Bayes model the TruncatedSVD was applied for dimensionality reduction on the training and testing data.
- This step helps in reducing the feature space while retaining essential information.
- The Gaussian Naive Bayes model was created and fitted to the transformed training data and the predictions were made on both the training and testing data.

c) Effectiveness of Algorithm

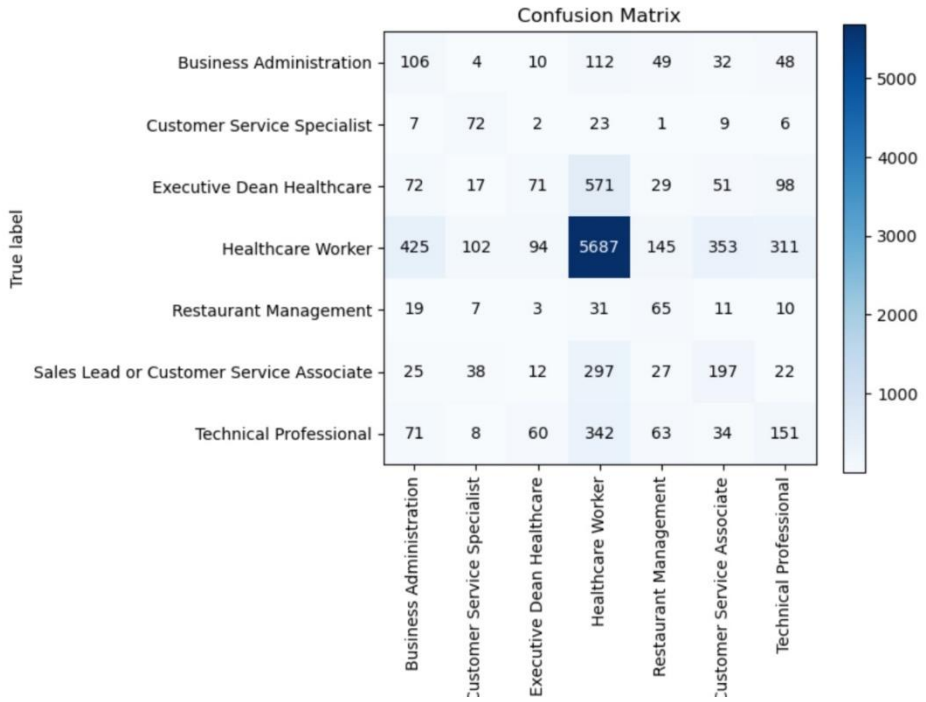
- Naive Bayes is computationally efficient and simple, making it suitable for fast setup and implementation.

Accuracy of Naive Bayes: 0.6324

	precision	recall	f1-score	support
Business Administration	0.14	0.29	0.19	361
Customer Service Specialist	0.30	0.58	0.40	120
Executive Dean Healthcare	0.28	0.07	0.12	909
Healthcare Worker	0.81	0.80	0.80	7117
Restaurant Management	0.15	0.42	0.23	146
Sales Lead or Customer Service Associate	0.28	0.33	0.31	618
Technical Professional	0.23	0.21	0.22	729
accuracy			0.63	10000
macro avg	0.31	0.39	0.32	10000
weighted avg	0.64	0.63	0.63	10000

d) Visualization

Out[21]: <Axes: title={'center': 'Confusion Matrix'}, xlabel='Predicted label', ylabel='True label'>



5) SUPPORT VECTOR MACHINES

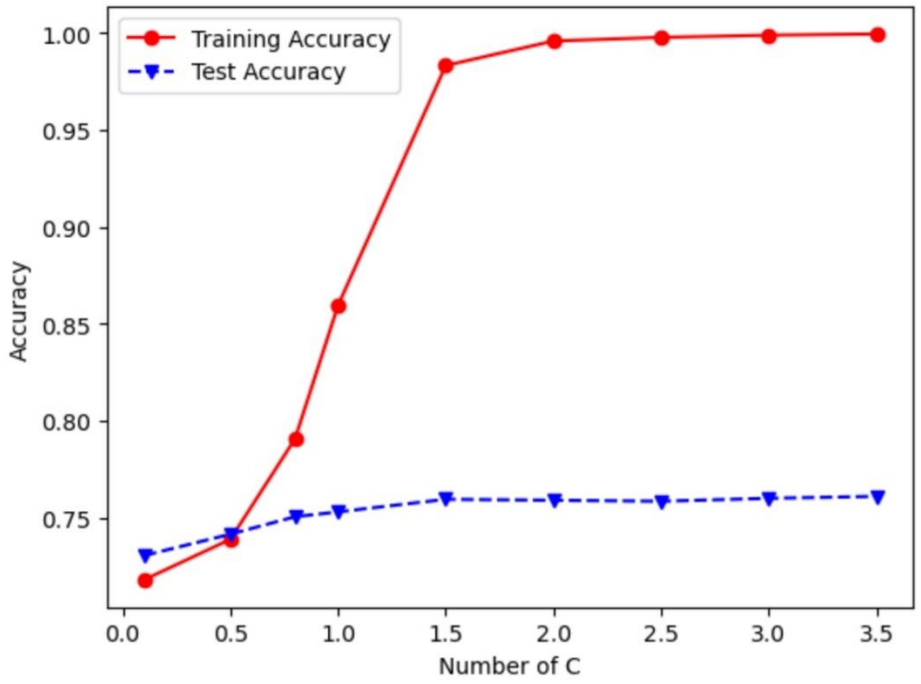
a) Why this algorithm?

- Margin Maximization: SVMs focus on finding the hyperplane with the maximum margin between positive (good fit) and negative (not a good fit) examples. This can lead to robust decision boundaries and potentially more accurate recommendations.

- Effective High-Dimensional Data Handling: SVMs excel at handling high-dimensional data, which is often the case with job recommendation systems where a user's skillset can be represented by many features.

b) Tuning and Training the model

- To tune and train the Support Vector Machine (SVM) classifier model, a range of C values (Csvm) is defined, which controls the trade-off between maximizing the margin and minimizing the classification error.
- Each C value is iterated over, creating an SVM classifier with the specified parameters: C value, gamma, and kernel function (in this case, 'rbf').
- The classifier is trained on the provided training data (X_train and y_train).



c) **Effectiveness of Algorithm**

- By handling non-linear relationships and maximizing margins, SVMs can potentially lead to more accurate recommendations compared to simpler models.

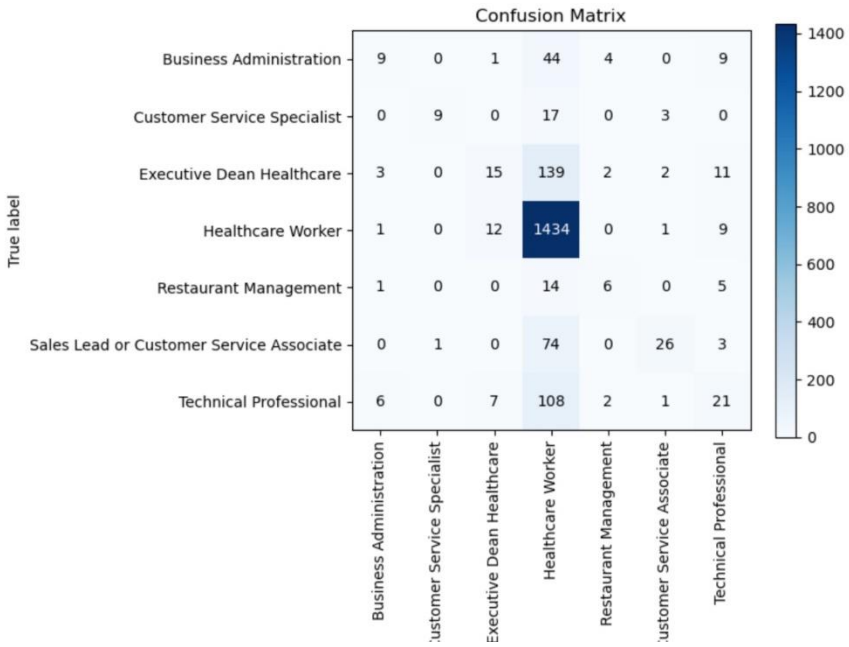
```
Accuracy of SVM: 0.76
                                    precision    recall  f1-score   support

                Business Administration       0.45      0.13      0.21        67
          Customer Service Specialist       0.90      0.31      0.46        29
          Executive Dean Healthcare       0.43      0.09      0.14       172
                  Healthcare Worker       0.78      0.98      0.87      1457
              Restaurant Management       0.43      0.23      0.30        26
Sales Lead or Customer Service Associate       0.79      0.25      0.38       104
              Technical Professional       0.36      0.14      0.21       145

                           accuracy                           0.76      2000
                          macro avg       0.59      0.31      0.37      2000
                       weighted avg       0.71      0.76      0.70      2000
```

d) Visualization

```
Out[67]: <Axes: title={'center': 'Confusion Matrix'}, xlabel='Predicted label', ylabel='True label'>
```



6) RANDOM FOREST CLASSIFICATION

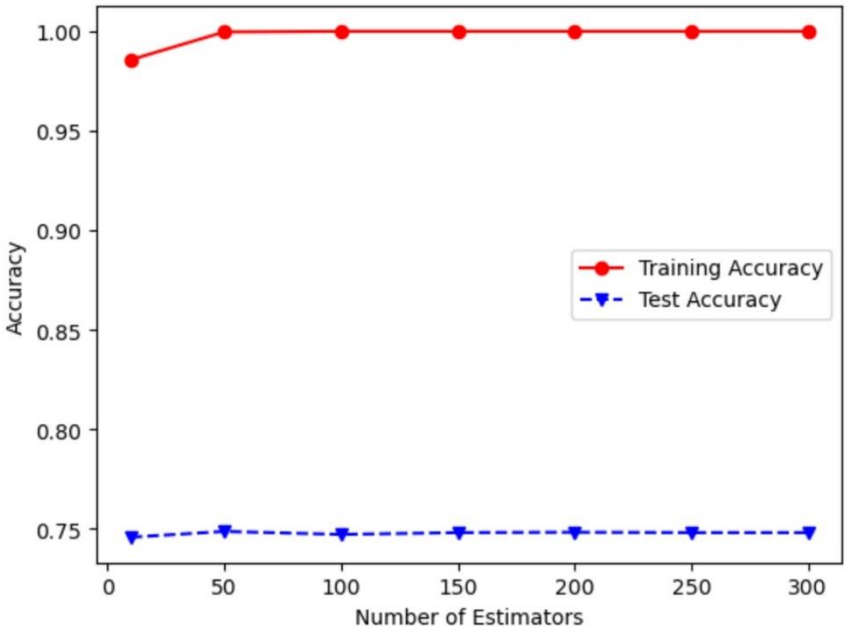
a) Why this algorithm?

- By handling non-linear relationships and maximizing margins, SVMs can potentially lead to more accurate recommendations compared to simpler models.

b) Tuning and Training the model

- To tune and train the Random Forest classifier model, a list n_estimators is initialized to contain different numbers of estimators, representing the number of trees in the forest.
- Iterating over each number of estimators, a RandomForestClassifier object is created with the specified number of trees and a random state for reproducibility.

- This classifier is then trained on the provided training data (X_train and y_train) and the predictions are generated for both the training and test datasets.



c) **Effectiveness of Algorithm**

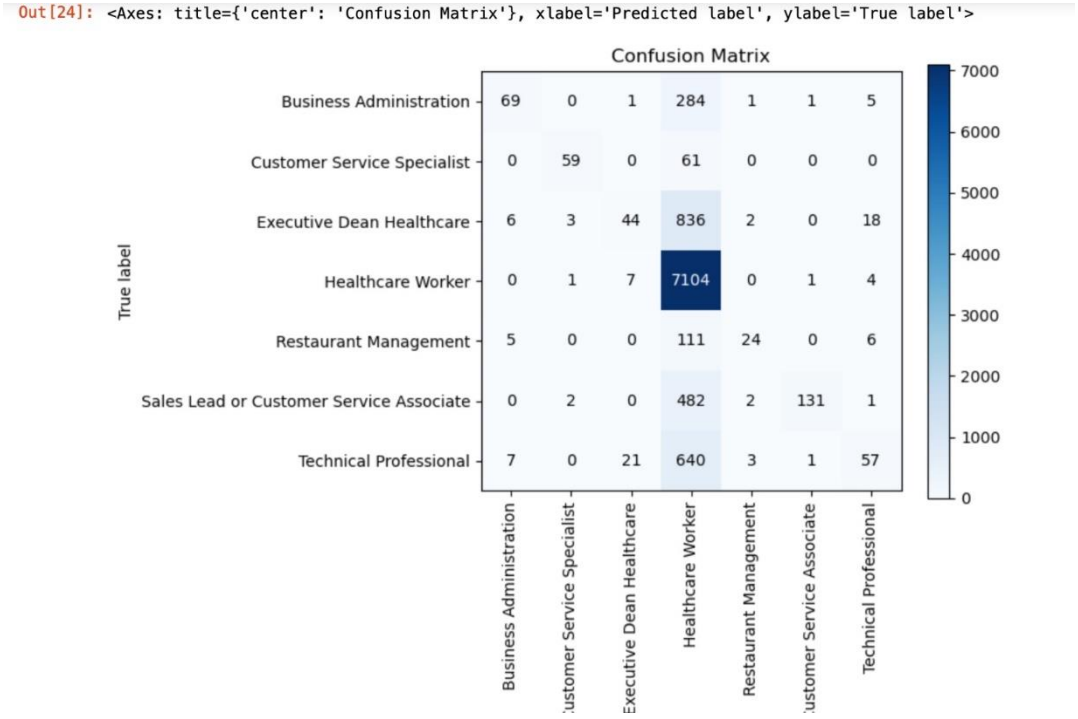
- By handling complex skill relationships and preventing overfitting, random forests can deliver more accurate recommendations compared to simpler models.

```
Accuracy of Random Forest: 0.7488
                                        precision    recall  f1-score   support

                Business Administration       0.79      0.19      0.31       361
             Customer Service Specialist       0.91      0.49      0.64       120
               Executive Dean Healthcare       0.60      0.05      0.09       909
                       Healthcare Worker       0.75      1.00      0.85      7117
                  Restaurant Management        0.75      0.16      0.27       146
Sales Lead or Customer Service Associate       0.98      0.21      0.35       618
                  Technical Professional       0.63      0.08      0.14       729

                                accuracy                           0.75     10000
                               macro avg       0.77      0.31      0.38     10000
                            weighted avg       0.74      0.75      0.67     10000
```

d) Visualization



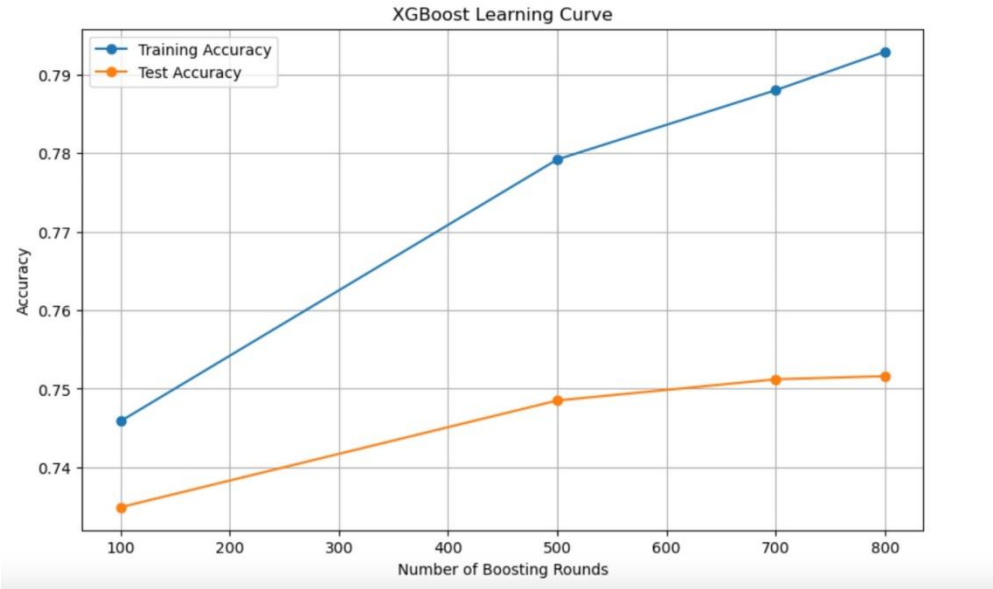
7. **XG-BOOST**

   a. **Why this algorithm?**

   - Scalability: XGBoost is efficient for handling large datasets with many users and jobs, making it suitable for real-world recommendation systems.
   - XGBoost is known for its high predictive accuracy. It is an ensemble learning method that combines the predictions from multiple individual models (typically decision trees) to produce a final prediction.

   b. **Tuning and Training the model**

   - To train the XGBoost classifier, the target variable is encoded using LabelEncoder to convert categorical labels into numerical values.
   - The parameters for the XGBoost model are defined, including the objective function, number of classes, maximum depth, regularization parameter (alpha), and learning rate.
   - A list of values for the number of boosting rounds (num_rounds_list) is specified and the XGBoost classifier is then iteratively trained with different numbers of boosting rounds.



c. **Effectiveness of Algorithm**

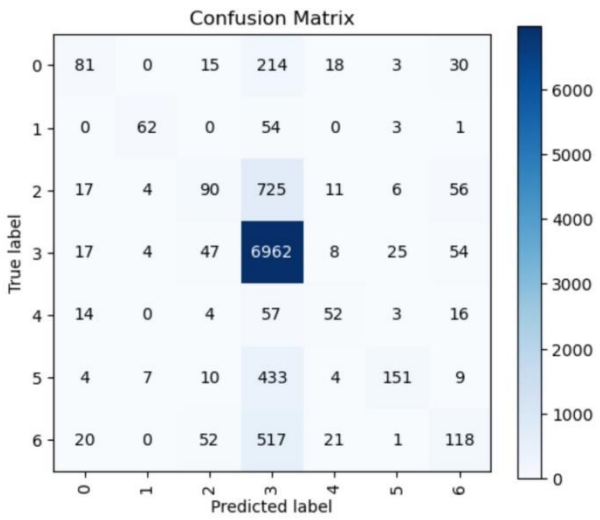
- By handling feature interactions and leveraging the ensemble approach, XGBoost can potentially generate more accurate and nuanced recommendations compared to simpler models.

Accuracy of XG-Boost: 0.7516

	precision	recall	f1-score	support
0	0.53	0.22	0.32	361
1	0.81	0.52	0.63	120
2	0.41	0.10	0.16	909
3	0.78	0.98	0.87	7117
4	0.46	0.36	0.40	146
5	0.79	0.24	0.37	618
6	0.42	0.16	0.23	729
accuracy			0.75	10000
macro avg	0.60	0.37	0.43	10000
weighted avg	0.70	0.75	0.70	10000

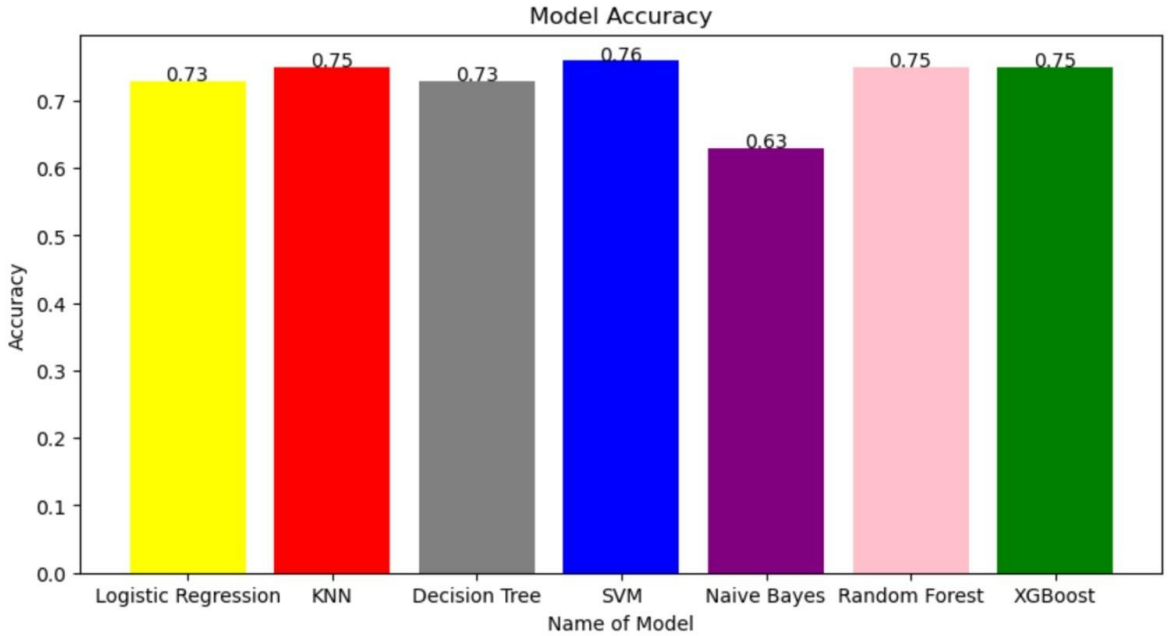
d. Visualization

<Axes: title={'center': 'Confusion Matrix'}, xlabel='Predicted label', ylabel='True label'>



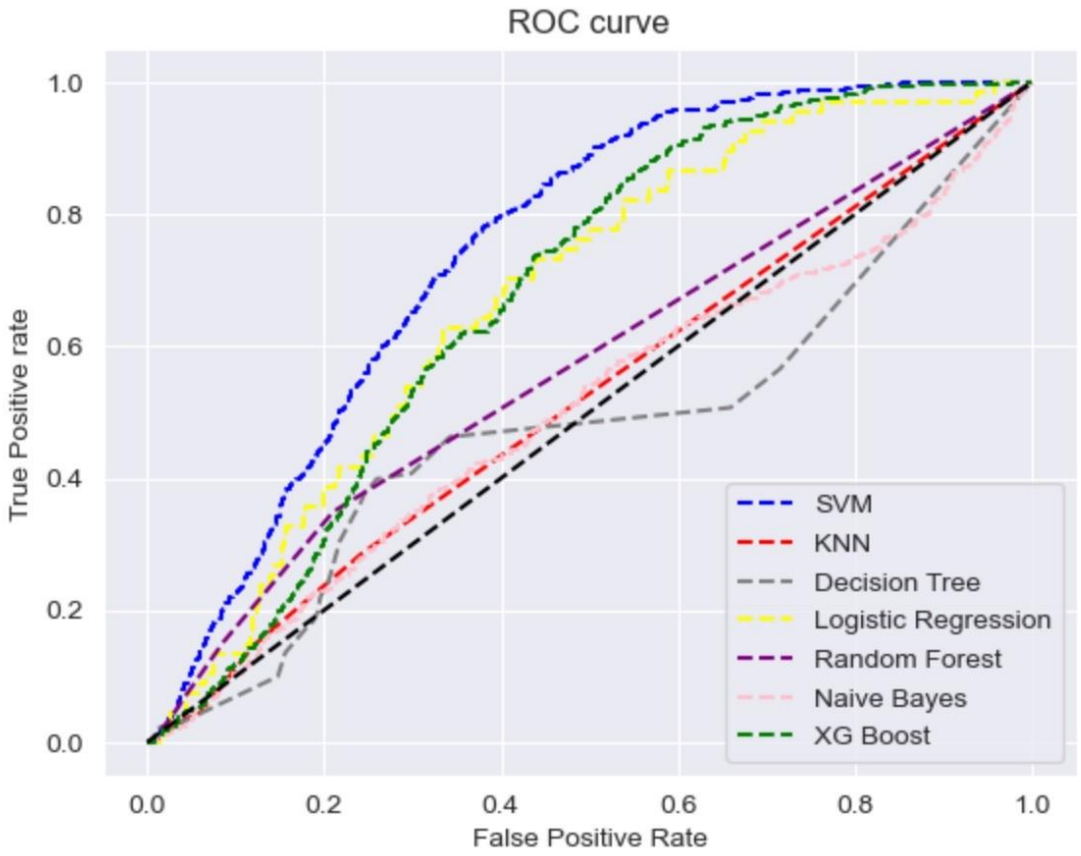
Comparison graph:
ACCURACY GRAPH

The accuracy of SVM is 76%(highest)
The accuracy of KNN, Random Forest and XGBoost is 75%
The accuracy of Logistic Regression and Decision Tree is 73%
The accuracy of Naive Bayes is 63%



ROC CURVE GRAPH

While we calculate the area under the curve(AUC), we can conclude that SVM is having the highest efficiency among the other algorithms for the given LinkedIn job postings dataset.



The Algorithm that has the highest efficiency and accuracy is SUPPORT VECTOR MACHINE algorithm

WORKING WITH THE BEST ALGORITHM(SVM):

Based on the user input for example, health and nurse, the job label is displayed. Then the jobs are identified with the label and the top 5 LinkedIn jobs along with the job title, url, location,etc are displayed.

```
Enter your skills: health, nurse
You may look into Healthcare Worker jobs
Here is a list of jobs that is under Healthcare Worker
```

Out[94]:

	Label	job_title	company_name	job_city	job_state	job_skills	job_level	job_post_link	last_modified_d
9154	Healthcare Worker	Travel RN - School RN	The Stepping Stones Group, LLC	Phoenix	AZ	School Nurse, Pediatric Nursing, Registered Nu...	Mid senior	https://www.linkedin.com/jobs/view/travel-rn-s...	24/01
2866	Healthcare Worker	Registered Nurse – Emergency Room - Travel - (...	TravelNurseSource	Southaven	MS	Emergency Room Nurse, Trauma Nurse, Acute Care...	Mid senior	https://www.linkedin.com/jobs/view/registered-...	24/01
6474	Healthcare Worker	RN - Health Educator	Spectrum Healthcare Resources	San Diego	CA	Health Promotion, Health Education, Program De...	Mid senior	https://www.linkedin.com/jobs/view/rn-health-e...	24/01
4705	Healthcare Worker	Occupational Health Nurse	Jobot	Sturgis	MI	Occupational Health Nursing, Registered Nurse ...	Mid senior	https://www.linkedin.com/jobs/view/occupationa...	24/01
5959	Healthcare Worker	Advanced Practice Nurse, Pediatric Psychiatry	Saint Anthony Hospital	Chicago	IL	Psychotherapy, Behavioral health home, Advance...	Mid senior	https://www.linkedin.com/jobs/view/advanced-pr...	24/01

Conclusion

We have taken the processed dataset from phase 1 and applied algorithms like SVM, Logistic Regression, Naïve Bayes, KNN, Decision Tree, XGBoost, Random Forest on it to find the best suited jobs for the skills given as input by the user. Based on the accuracy and ROC for each algorithms we can conclude that SVM is the best suited algorithm among the 7 for the job recommendation system.

REFERENCE

The algorithms are taken from the following references:

- https://scikit-learn.org/stable/
- https://xgboost.readthedocs.io/en/stable/
- Book: Data Science From Scratch - First Principles with Python by Joel Grus

Project Phase #3

Team Member 1: DHIVYASHREE SIVAPRAKASAM (UBIT – dsivapra, Person number - 50560688)

Team Member 2: SHALINI ANANTHAVEL JAYALAKSHMI (UBIT – ananthav, Person number - 50560497)

Team Member 3: GANESH PRABAKARAN (UBIT – ganeshpr, Person number - 50560751)

PROBLEM STATEMENT:

The exponential growth of online job portals and professional networking platforms has revolutionized the recruitment process by providing individuals with convenient access to job listings and networking opportunities. However, the sheer volume of available job postings often results in information overload, making it difficult for job seekers to identify relevant opportunities. Moreover, traditional job search methods often lack personalization and fail to consider individual preferences, skills, and career aspirations. Consequently, there is a pressing need for an intelligent system that can analyze user data and provide personalized job recommendations tailored to the unique profile of each user.

The project seeks to solve the problems of information overload, lack of personalization and inefficiency in the job-hunting process by developing a personalized job recommendation system that leverages user data and machine learning algorithms to deliver tailored job suggestions.

Process Documentation:

Requirement:

Python and VS Code

Steps to run the application:

1. Extract the zip file and open the folder in VS Code
2. Set the Python Interpreter (ctrl + shift +p, search “python interpreter” and select the respective python in your system)
3. Install python extension in VS Code from extensions
4. If there venv folder is present delete it before running the commands
5. Now, give the following commands:

python -m venv venv- creates a virtual environment named 'venv' in the current directory.

.\venv\Scripts\activate – Activate the virtual environment.

(If you find any problem in this in windows system open powershell in admin mode and give the command: Set-ExecutionPolicy Unrestricted -Force)

pip install numpy – Install numpy package

pip install numpy pandas– Install numpy pandas package

pip install flask – Installs Flask package

pip install scikit-learn – Installs machine learning library scikit

6. Finally, give the below command to run the application:

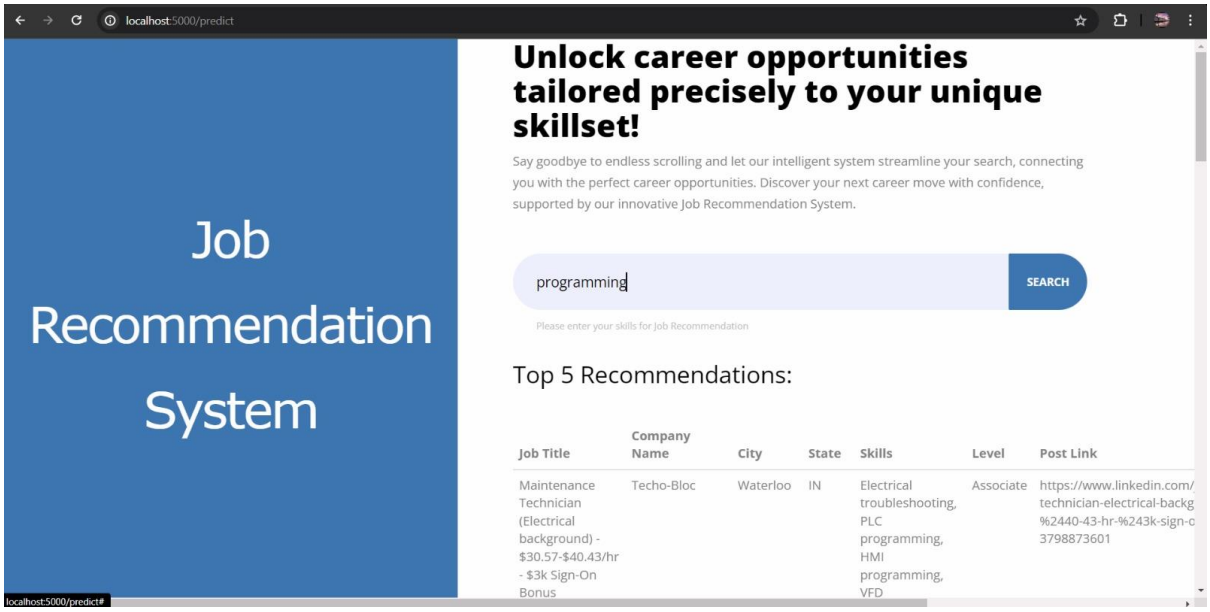
python app.py – Runs the script which contains code for flask application

7. Open localhost:5000 in the web browser to view the application running
8. Give skill as input in the text box and click on search button:

Sample input:

Programming

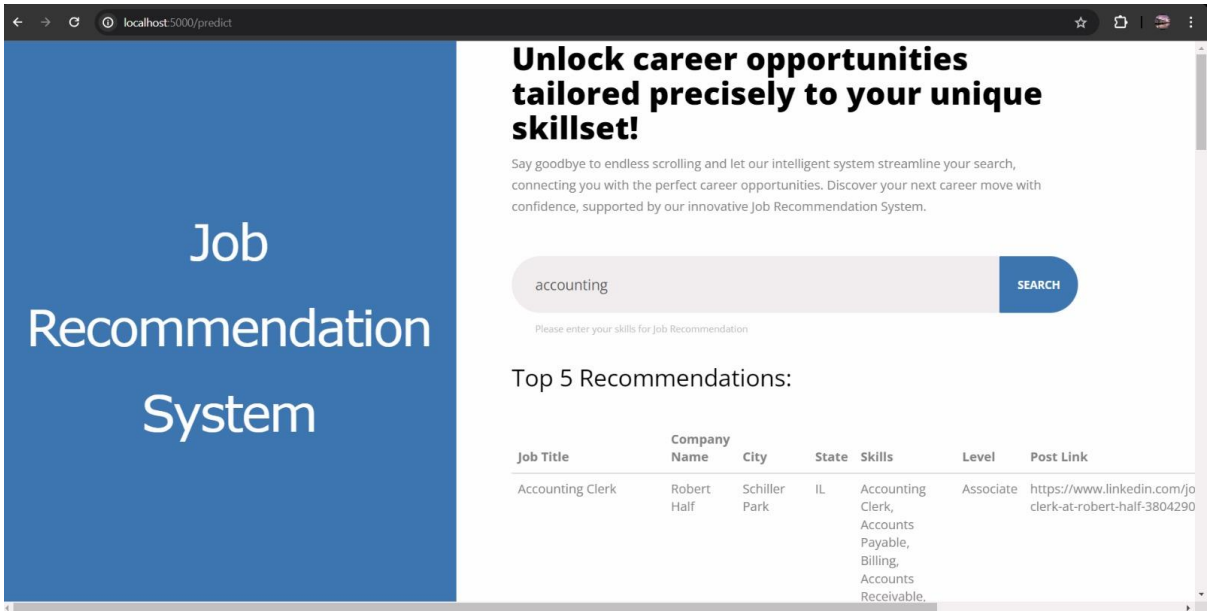
Output:



Sample input:

accounting

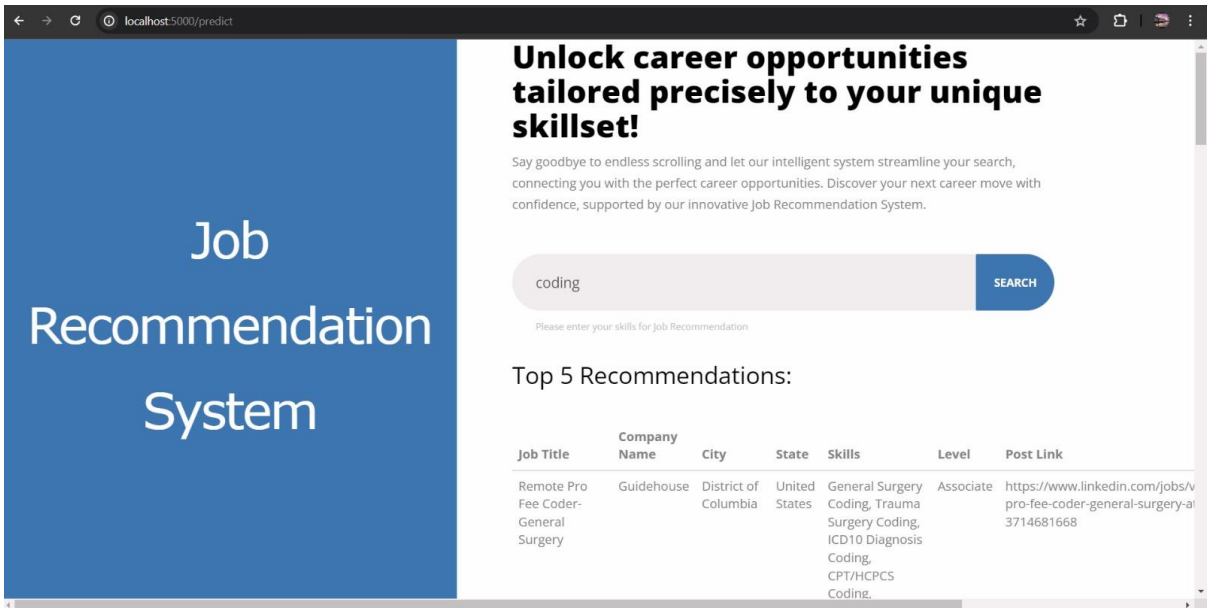
Output:



Sample input:

coding

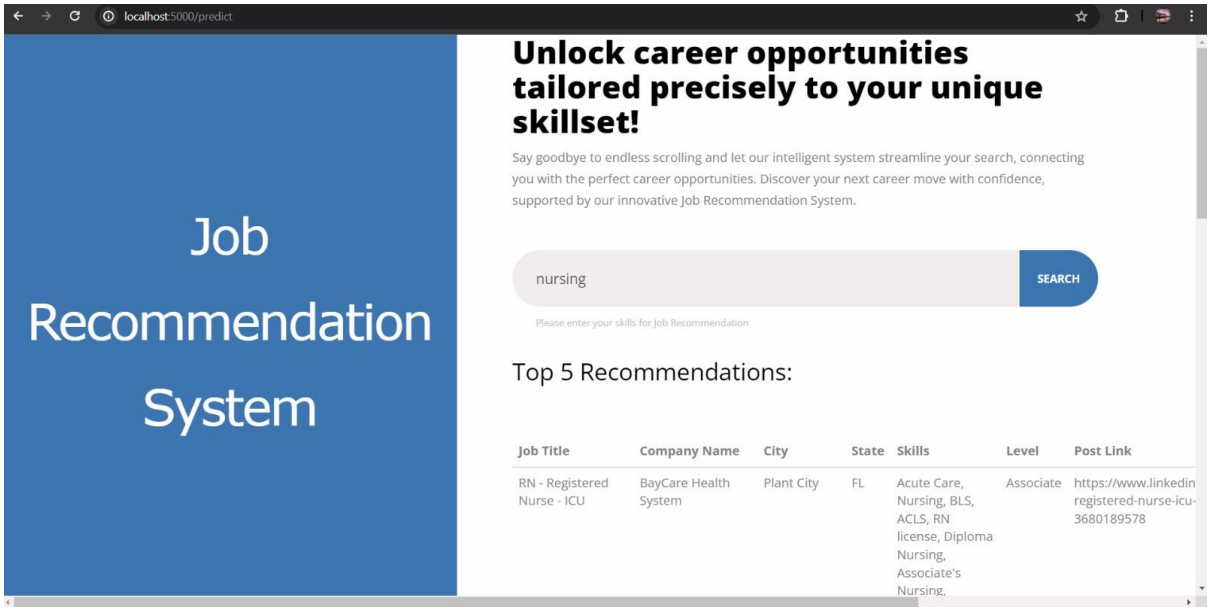
Output:



Sample input:

nursing

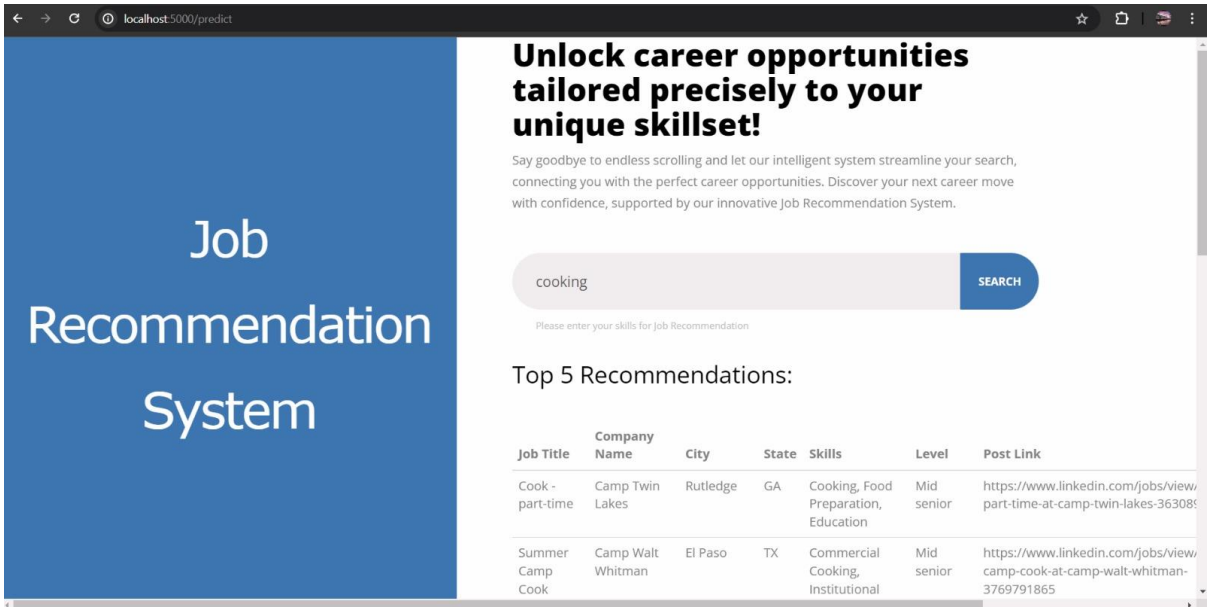
Output:



Sample input:

cooking

Output:



9. Now the recommended top 5 jobs along with the linkedIn url, location, etc will be displayed in the table below

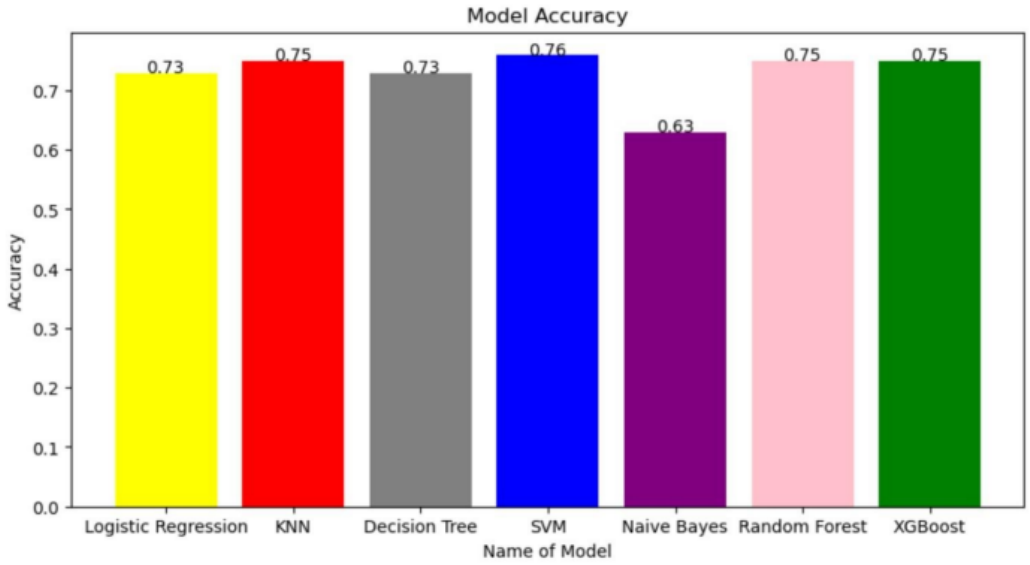
ACCURACY GRAPH

The accuracy of SVM is 76%(highest)

The accuracy of KNN, Random Forest and XGBoost is 75%

The accuracy of Logistic Regression and Decision Tree is 73%

The accuracy of Naive Bayes is 63%



The Algorithm that has the highest efficiency and accuracy is SUPPORT VECTOR MACHINE algorithm

Algorithm Used:

Support Vector Machine

Based on the user input for example, health and nurse, the job label is displayed.

Then the jobs are identified with the label and the top 5 LinkedIn jobs along with the job title, url, location,etc are displayed. For tuning the model these values Csvm = [0.1,0.5,0.8,1,1.5,2,2.5,3,3.5], C=5, gamma=1 are used.

```
Enter your skills: health, nurse
You may look into Healthcare Worker jobs
Here is a list of jobs that is under Healthcare Worker
```

Out[94]:

	Label	job_title	company_name	job_city	job_state	job_skills	job_level	job_post_link	last_modified_d
9154	Healthcare Worker	Travel RN - School RN	The Stepping Stones Group, LLC	Phoenix	AZ	School Nurse, Pediatric Nursing, Registered Nu...	Mid senior	https://www.linkedin.com/jobs/view/travel-rn-s...	24/01
2866	Healthcare Worker	Registered Nurse – Emergency Room - Travel - (...	TravelNurseSource	Southaven	MS	Emergency Room Nurse, Trauma Nurse, Acute Care...	Mid senior	https://www.linkedin.com/jobs/view/registered-...	24/01
6474	Healthcare Worker	RN - Health Educator	Spectrum Healthcare Resources	San Diego	CA	Health Promotion, Health Education, Program De...	Mid senior	https://www.linkedin.com/jobs/view/rn-health-e...	24/01
4705	Healthcare Worker	Occupational Health Nurse	Jobot	Sturgis	MI	Occupational Health Nursing, Registered Nurse ...	Mid senior	https://www.linkedin.com/jobs/view/occupationa...	24/01
5959	Healthcare Worker	Advanced Practice Nurse, Pediatric Psychiatry	Saint Anthony Hospital	Chicago	IL	Psychotherapy, Behavioral health home, Advance...	Mid senior	https://www.linkedin.com/jobs/view/advanced-pr...	24/01

Web Development:

We have used Flask micro web framework to create the web application.

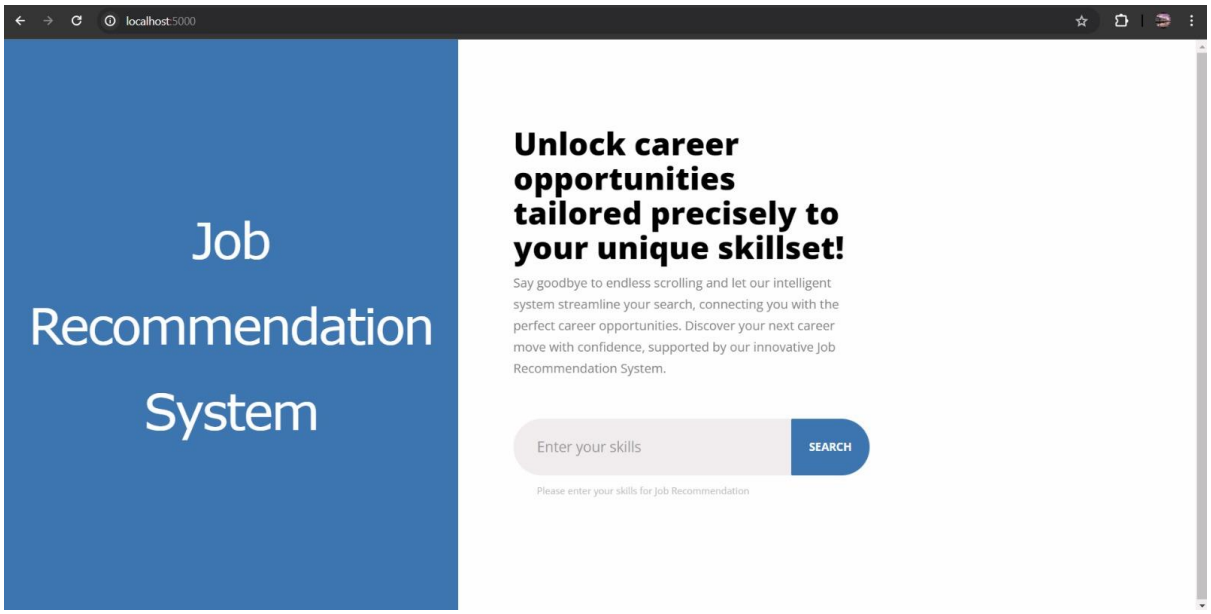
Three .pkl files are generated (svm_model, vectorizer and job_data)

These are loaded in the app.py. Here we got the skills from the user as a text from the text box in index.html(which is present under templates folder).

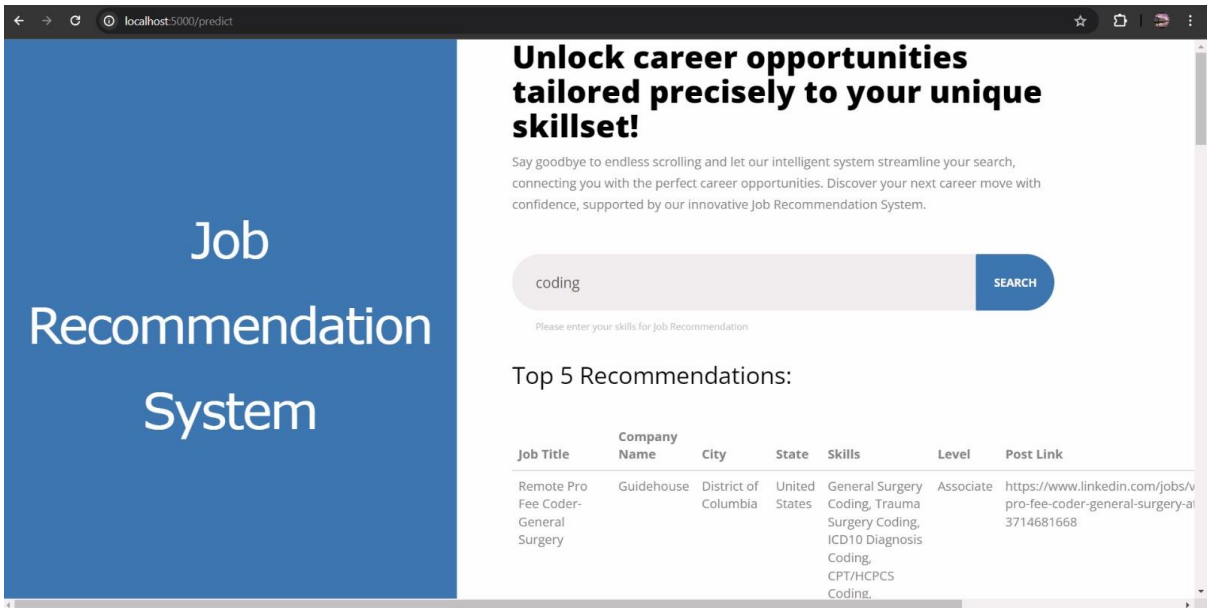
The css, bootstrap, js and vendor folders are present inside the static folder.

The /predict is used as a POST call to be called once the user clicks the search button and the output will be displayed as a table below the text box div.

Localhost:5000



Localhost:5000/predict



Users can learn several valuable things from the product you've developed:

- **Optimal Job Matches**: Users can understand which job opportunities are best suited to their skills and qualifications. By inputting their skills into the system, they receive personalized recommendations based on machine learning algorithms.
- **Personalization Beyond Keywords:** Unlike traditional keyword-based searches, which may yield generic results, the machine learning-based approach offers tailored suggestions by analyzing intricate patterns in user data.
- **Continuous Improvement:** Users can recognize the value of systems that continuously learn and adapt based on user feedback and behavior.

Extensions or avenues for exploration could include:

- Leveraging NLP techniques to better understand user queries and job descriptions could improve the matching process.
- Implementing robust mechanisms for collecting and analysing user feedback can further refine the recommendation system and provide valuable insights for future iterations.

Conclusion

- We have taken the processed dataset from phase 1 and applied algorithms like SVM, Logistic Regression, Naïve Bayes, KNN, Decision Tree, XGBoost, Random Forest on it in Phase 2 to find the best suited jobs for the skills given as input by the user and developed an web application for the same in Phase 3.
- Based on the accuracy and ROC for each algorithms we can conclude that SVM is the best suited algorithm among the 7 for the job recommendation system and we have used that to build the personalised job recommendation system.
- Personalized recommendations that go beyond traditional keyword-based searches.
- While key-based systems rely on predefined rules and may struggle with scalability and bias, Machine learning offers adaptive and nuanced job recommendations by analyzing complex patterns and user data.
- These systems continually learn and adapt based on user feedback and behavior to improve the accuracy of their recommendations over time.