Project Writeup: LinkedIn Job Analysis

ENPM808W

Divyansh Shrivastava
Sravya Lenka
Surbhi Garg

Introduction

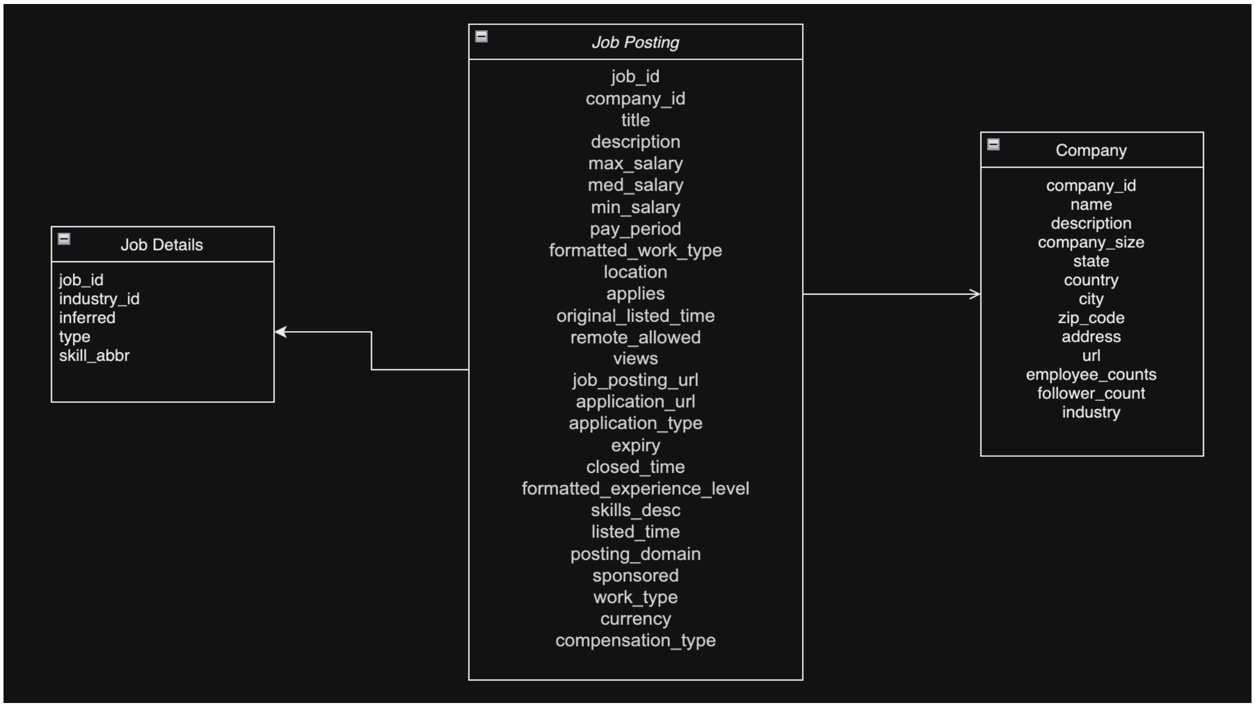
In the ever-evolving landscape of today's job market, staying ahead of employment trends is not just beneficial, but essential for both job seekers and employers. Our initiative, titled "LinkedIn Job Analysis," is rooted in this very understanding. We embarked on a mission to harness the wealth of data from LinkedIn Job Postings, aiming to extract and distill comprehensive insights into the currents shaping the job market today.

As International Students who have personally navigated the daunting task of applying to hundreds of jobs, we recognize the profound importance of understanding job market trends and the level of competition within it. This insight is particularly crucial for students like us, who must continually adapt to the fluctuating dynamics of the job market. Our project, therefore, is more than an academic exercise; it's a practical solution crafted from our own experiences and needs.

Through our analysis, we aimed to go beyond mere observation, providing predictive insights on the number of applicants for future job postings. This aspect of our project is designed to empower job seekers with foresight, enabling them to anticipate competition and prepare accordingly. Similarly, for employers, these insights offer a window into the demand dynamics of various roles, assisting them in understanding how to attract the right talent.

Data

Our initial idea was to scrape the data from LinkedIn. Though we were successfully able to scrape the data, given the project timeframe, we would not have data spread out across various periods which was important for us. So, we sourced our dataset from Kaggle, opting for the latest job posting data due to its alignment with our project goals. The dataset was not only sizable but also well-organized, meeting our criteria for a robust analysis. It has 15,886 rows with 27 columns. We possessed supplementary information regarding the job postings, including details about the company and job benefits. The structure of the data is as follows:



Job Posting

job_id
company_id
title
description
max_salary
med_salary
min_salary
pay_period
formatted_work_type
location
applies
original_listed_time
remote_allowed
views
job_posting_url
application_url
application_type
expiry
closed_time
formatted_experience_level
skills_desc
listed_time
posting_domain
sponsored
work_type
currency
compensation_type

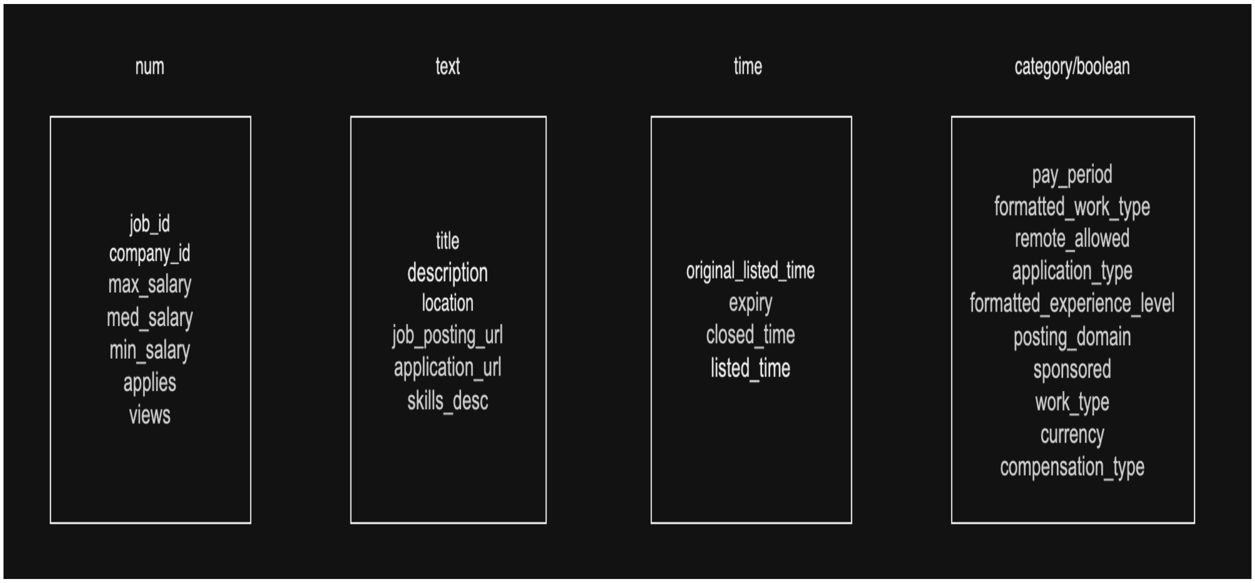
Job Details

job_id
industry_id
inferred
type
skill_abbr

Company

company_id
name
description
company_size
state
country
city
zip_code
address
url
employee_counts
follower_count
industry

Our dependent variable is **'applies'** (which indicates the number of people who applied for the job) from job posting data and we started with understanding the available job posting data and how these features influence our dependent variable and the data available.



num

job_id
company_id
max_salary
med_salary
min_salary
applies
views

text

title
description
location
job_posting_url
application_url
skills_desc

time

original_listed_time
expiry
closed_time
listed_time

category/boolean

pay_period
formatted_work_type
remote_allowed
application_type
formatted_experience_level
posting_domain
sponsored
work_type
currency
compensation_type

Project and Process

We divided our project into standard phases, encompassing exploratory data analysis (EDA), data cleaning, feature engineering, modeling, and evaluation.

Division of work

Acquiring Data - Divyansh, Sravya
Data Cleaning - Sravya, Surbhi
Exploratory Data Analysis - Surbhi, Divyansh
Feature Engineering - Divyansh, Sravya
Model & Evaluation Metrics - Sravya, Surbhi
Hyperparamter Tuning - Surbhi
Documentation & Presentation - Divyansh, Sravya, Surbhi

Data Cleaning and EDA

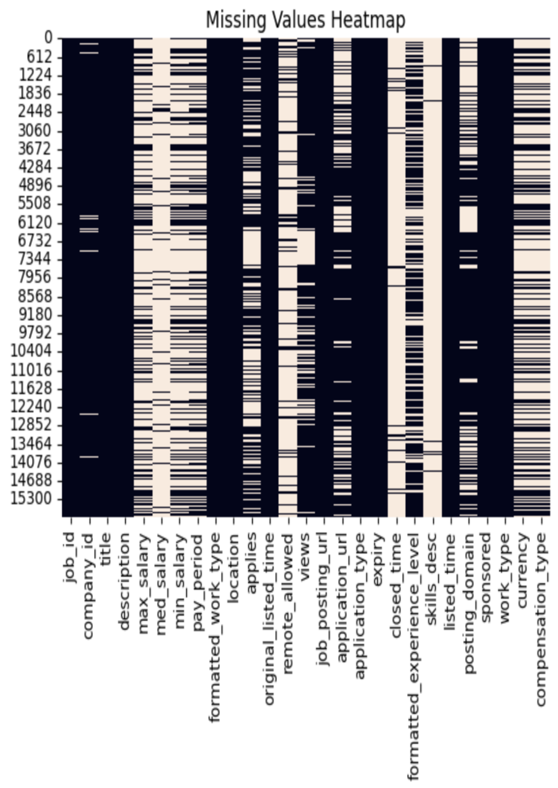
We tackled Data Cleaning and Exploratory Data Analysis (EDA) simultaneously to grasp the data's nuances and ensure it was ready for our model.

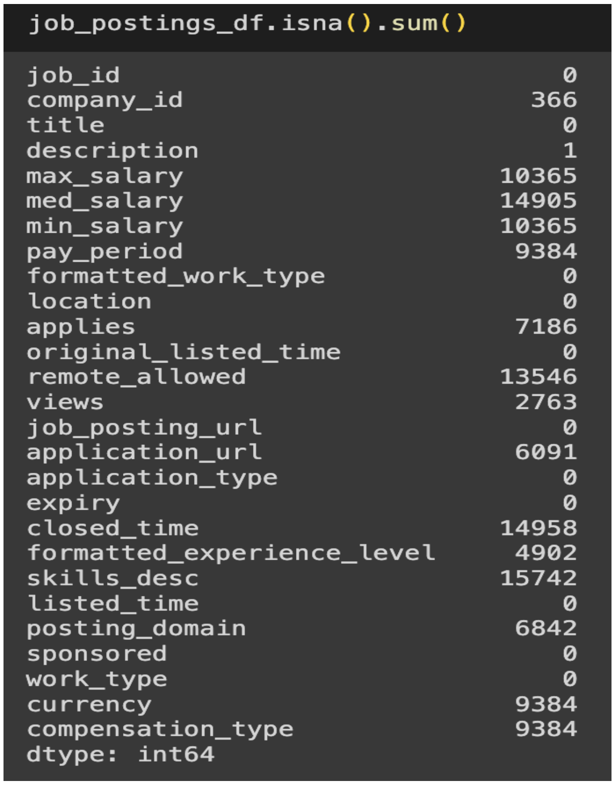
In the Data cleaning stage, we meticulously worked on “initial data assessment”, which involved evaluating the overall structure, identifying data types for each column, and detecting and dealing with inconsistencies and anomalies in the data. We also did “Geographical Data Extraction”, where we extracted and standardized state information from the ‘location’ field, enhancing the granularity of our geographical analysis. We also focused on “timestamp standardization” where we converted multiple timestamp columns to a uniform datetime format to provide consistency and ease of use in our project.

In the EDA phase, we delved into column values, checked correlations, and explored categorical columns' influence on applications received. We addressed missing values strategically, dropping rows for the dependent variable "applies" and performing data imputation where necessary.

Understanding Missing Values:

Understanding the missing values was important in structuring our data. Here is our initial assessment of the missing values:

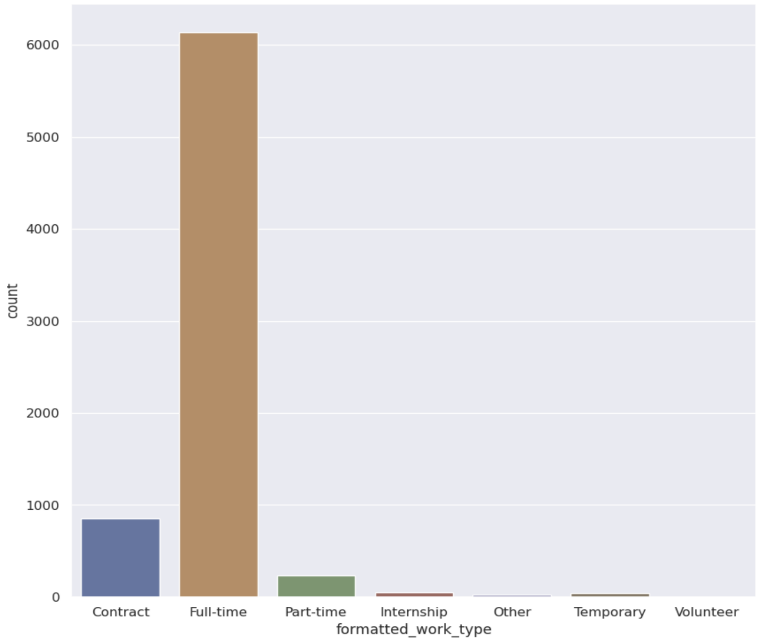




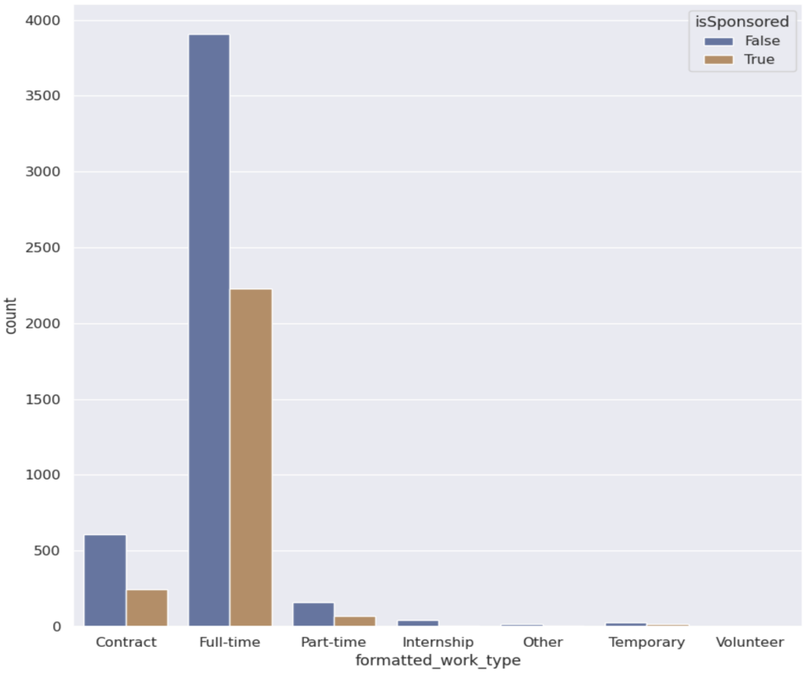
```
job_postings_df.isna().sum()

job_id                          0
company_id                    366
title                           0
description                     1
max_salary                  10365
med_salary                  14905
min_salary                  10365
pay_period                   9384
formatted_work_type             0
location                        0
applies                      7186
original_listed_time            0
remote_allowed              13546
views                        2763
job_posting_url                 0
application_url              6091
application_type                0
expiry                          0
closed_time                 14958
formatted_experience_level   4902
skills_desc                 15742
listed_time                     0
posting_domain               6842
sponsored                       0
work_type                       0
currency                     9384
compensation_type            9384
dtype: int64
```

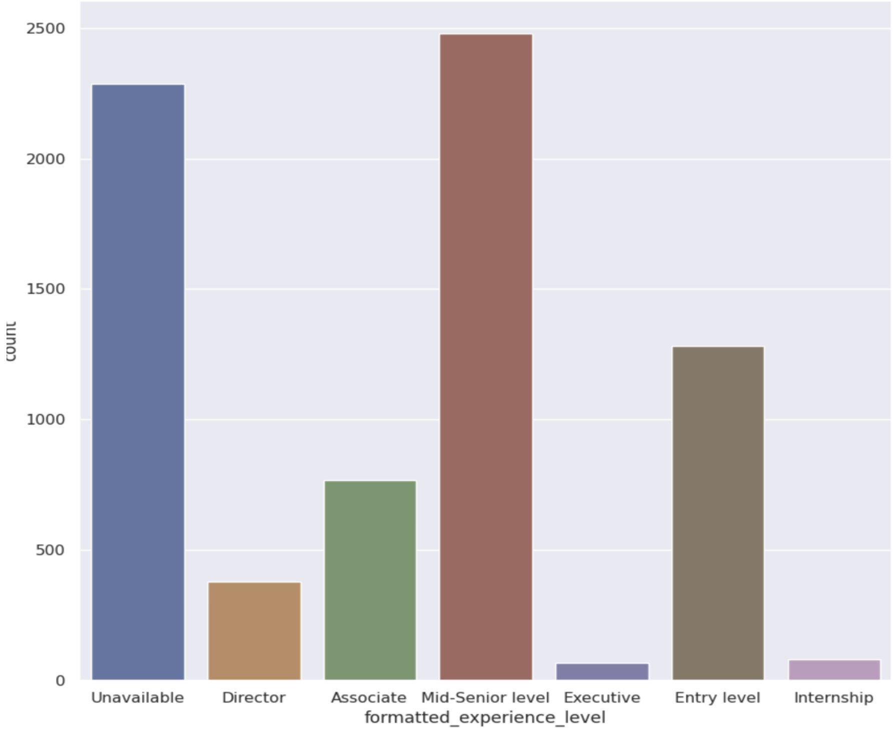
Work type: As we tried to understand the role of work type (which is a categorical variable), it is clear from the chart that most of the data we possessed belonged to Worktype - Full Time.



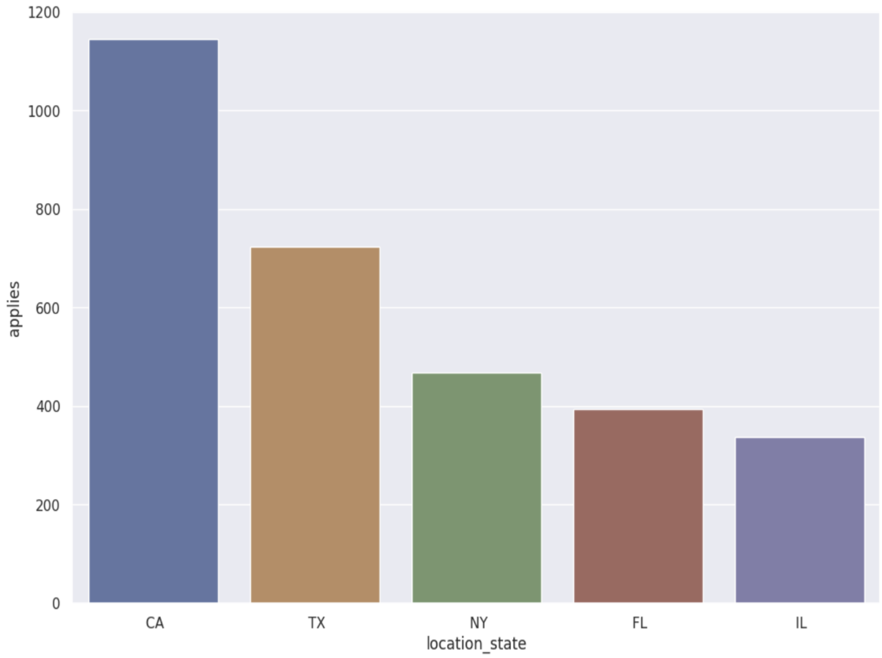
Work Type with Sponsorship: Based on the data we had, the majority of the jobs were not sponsored but a chunk of FTE roles compared to other work types are Sponsored.



Experience Level: Mid-senior and Entry-level are the most sought-after experience levels.

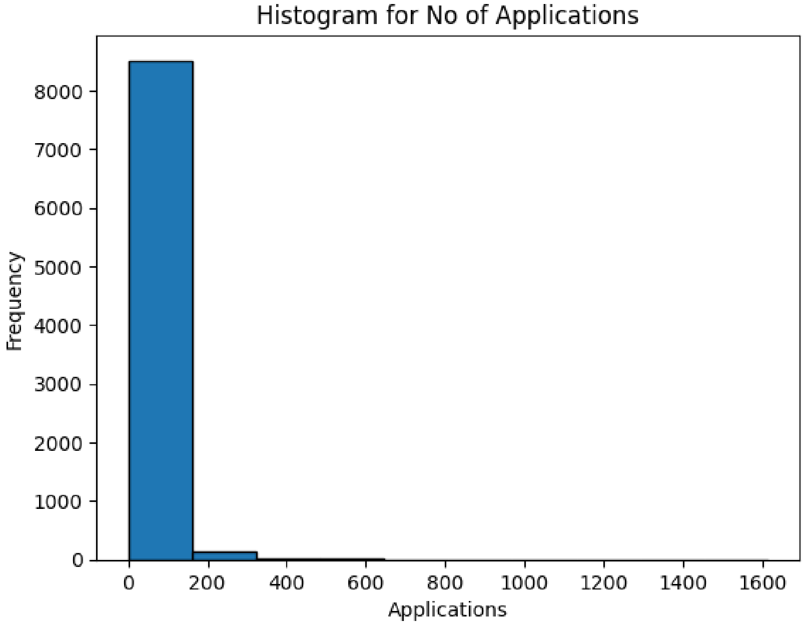


State: California, Texas, and New York are the states with the most number of jobs, coming close were Florida and Illinois.



Understanding dependent variable - ‘applies’

Most of the job applications have applications in the 0-200 range.

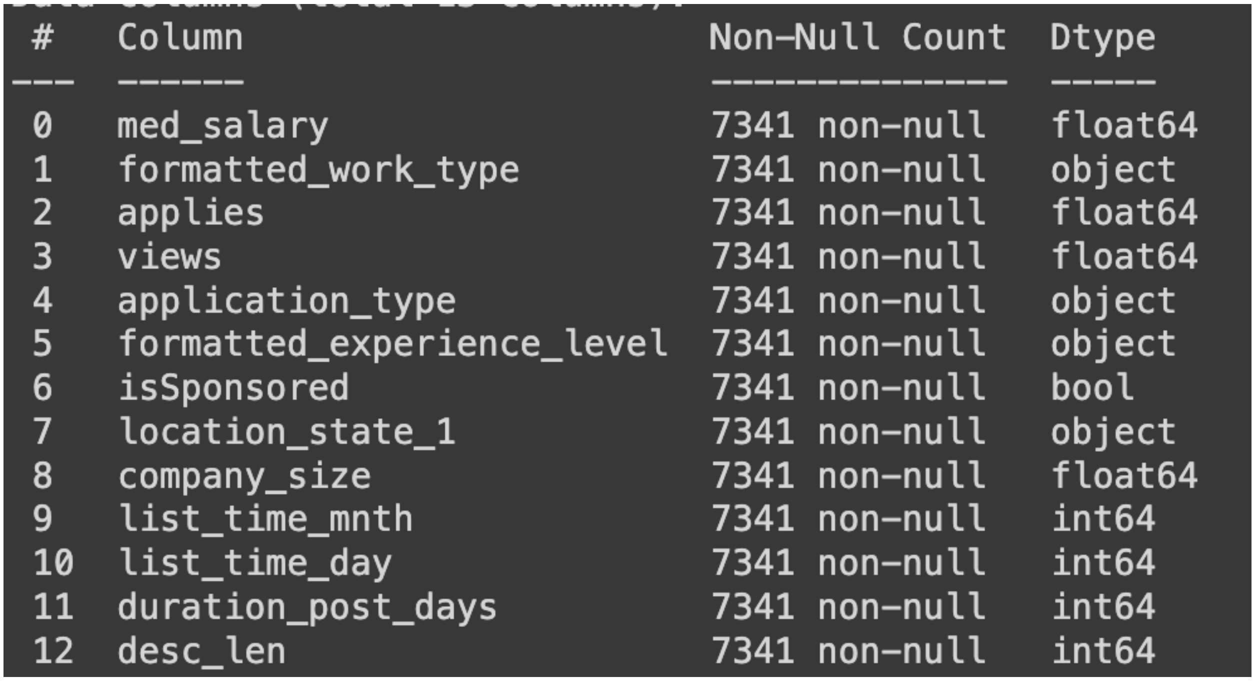


Feature Engineering and Selection

Feature engineering enhanced our modeling approach, involving one-hot encoding of categorical variables, creating new features like "desc_len" based on job description length, extracting the day and month of job postings, and calculating job posting duration. Column streamlining and redundancy removal was a big part of this stage as we eliminated redundant columns like ‘closed_time’, and ‘original_listed_time’ streamlining the dataset for focused analysis and reducing computational complexity. New features were created to quantify the ‘listed_time’ in various units such as days and months, offering nuanced insights into the timing aspects of job postings. Apart from this, the difference between ‘listed_time’ and ‘expiry’ was also accounted for as we suspected that the number of days a job posting was valid had an influence on the applications.

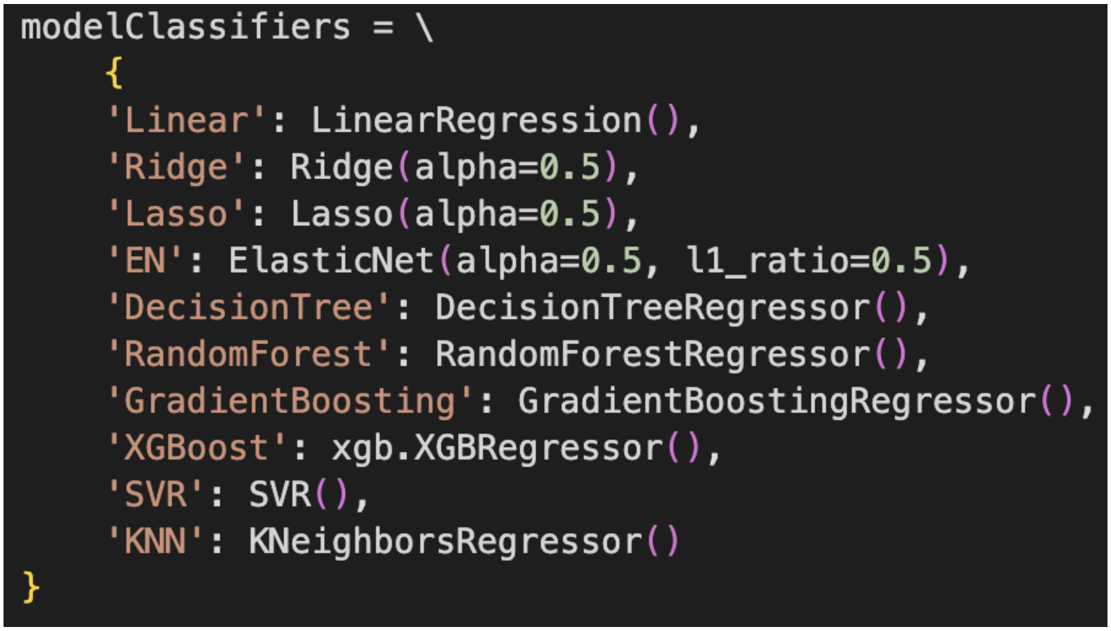
Models

We divided the data into train and test data where 75% data was used for training and 25% data used for testing. We tested with a varied set of features but our best models are with the following features -



```
 #   Column                     Non-Null Count  Dtype  
---  ------                     --------------  -----  
 0   med_salary                 7341 non-null   float64
 1   formatted_work_type        7341 non-null   object 
 2   applies                    7341 non-null   float64
 3   views                      7341 non-null   float64
 4   application_type           7341 non-null   object 
 5   formatted_experience_level 7341 non-null   object 
 6   isSponsored                7341 non-null   bool   
 7   location_state_1           7341 non-null   object 
 8   company_size               7341 non-null   float64
 9   list_time_mnth             7341 non-null   int64  
 10  list_time_day              7341 non-null   int64  
 11  duration_post_days         7341 non-null   int64  
 12  desc_len                   7341 non-null   int64  
```

Then we trained ten models, including Linear Regression, Decision Tree, Random Forest, and XGBoost on the training data and evaluated the test data. The models we used are in the following picture:

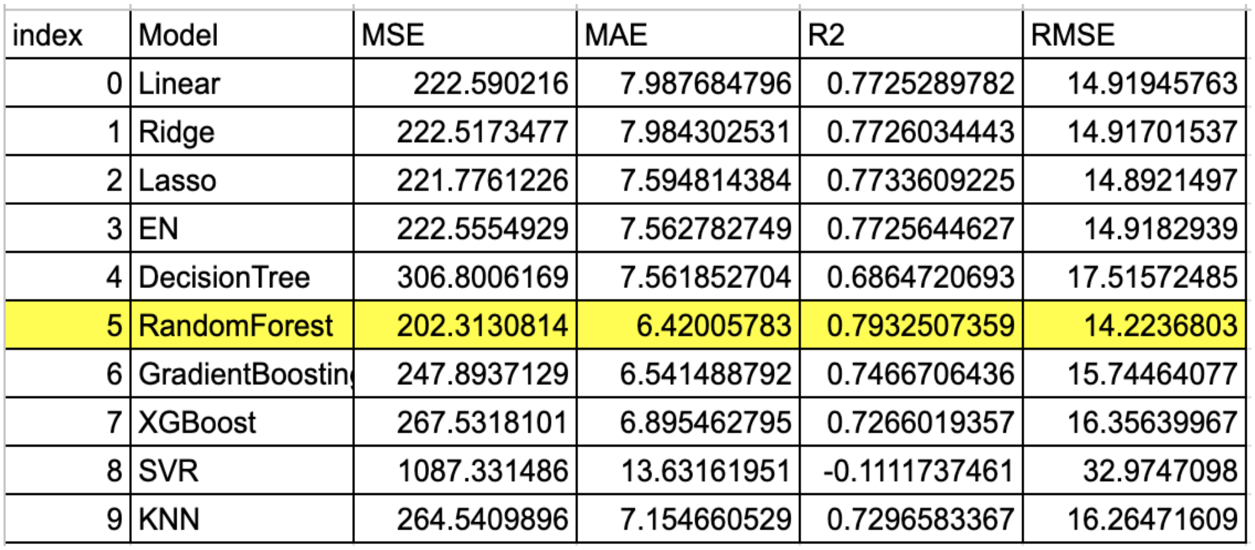


```
modelClassifiers = \
    {
    'Linear': LinearRegression(),
    'Ridge': Ridge(alpha=0.5),
    'Lasso': Lasso(alpha=0.5),
    'EN': ElasticNet(alpha=0.5, l1_ratio=0.5),
    'DecisionTree': DecisionTreeRegressor(),
    'RandomForest': RandomForestRegressor(),
    'GradientBoosting': GradientBoostingRegressor(),
    'XGBoost': xgb.XGBRegressor(),
    'SVR': SVR(),
    'KNN': KNeighborsRegressor()
}
```

The evaluation was based on RMSE and R2 scores. Notably, Random Forest emerged as the best-performing model with an RMSE score of 14.55 and an R2 score of 0.78.

Metrics

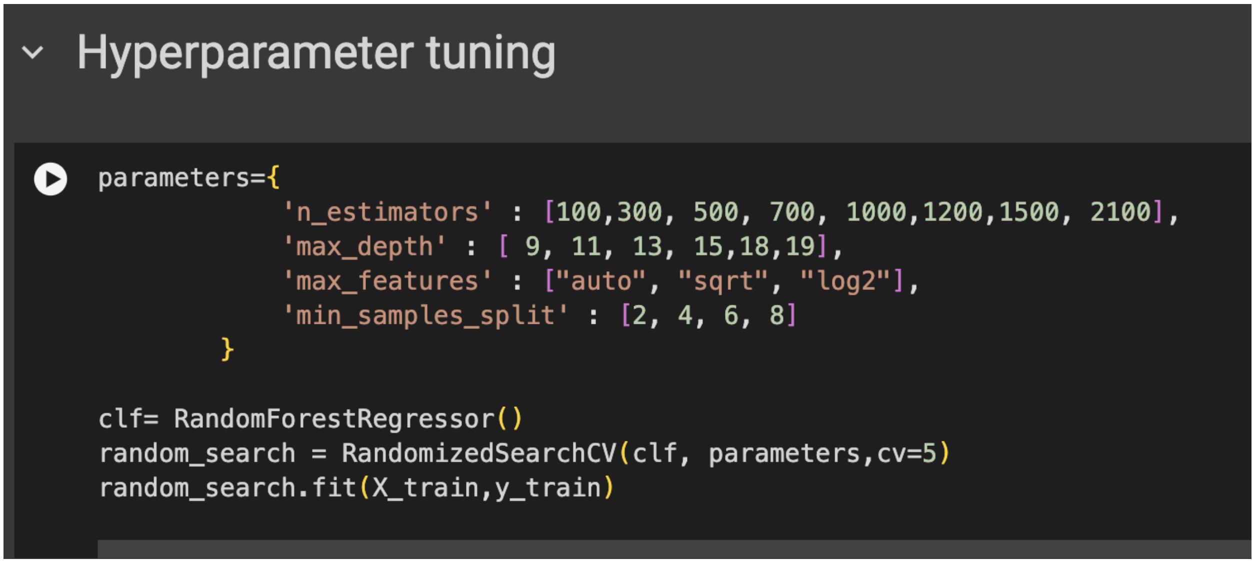
To evaluate our models, though we calculated Mean Squared Error(MSE), Mean Absolute Error(MAE), R-squared(R2), and Root Mean Squared Error(RMSE) - we took a combination of R2 and RMSE so that we could choose a model that is better with both the fit of the data and the predictions.



index	Model	MSE	MAE	R2	RMSE
0	Linear	222.590216	7.987684796	0.7725289782	14.91945763
1	Ridge	222.5173477	7.984302531	0.7726034443	14.91701537
2	Lasso	221.7761226	7.594814384	0.7733609225	14.8921497
3	EN	222.5554929	7.562782749	0.7725644627	14.9182939
4	DecisionTree	306.8006169	7.561852704	0.6864720693	17.51572485
5	RandomForest	202.3130814	6.42005783	0.7932507359	14.2236803
6	GradientBoostin	247.8937129	6.541488792	0.7466706436	15.74464077
7	XGBoost	267.5318101	6.895462795	0.7266019357	16.35639967
8	SVR	1087.331486	13.63161951	-0.1111737461	32.9747098
9	KNN	264.5409896	7.154660529	0.7296583367	16.26471609

Hyperparameter Tuning & Final Model

In this phase, we delved deeper into the realm of model optimization, a process pivotal for enhancing the predictive accuracy and reliability of our analysis. Recognizing the importance of fine-tuning in machine learning, we employed the robust technique of Random Grid Search (RandomGridSearchCV). This powerful method systematically navigates through a predefined range of hyperparameter values, searching for the combination that yields the most effective model performance.



Hyperparameter tuning

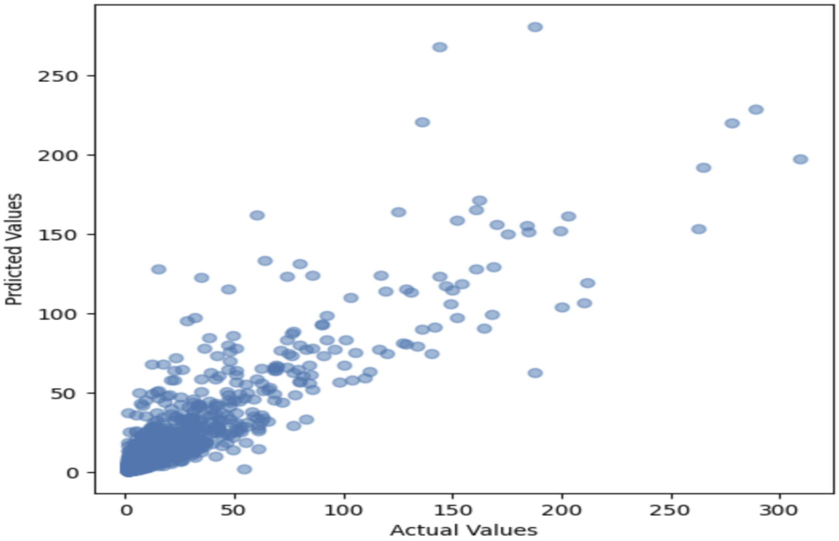
```
parameters={
            'n_estimators' : [100,300, 500, 700, 1000,1200,1500, 2100],
            'max_depth' : [ 9, 11, 13, 15,18,19],
            'max_features' : ["auto", "sqrt", "log2"],
            'min_samples_split' : [2, 4, 6, 8]
        }

clf= RandomForestRegressor()
random_search = RandomizedSearchCV(clf, parameters,cv=5)
random_search.fit(X_train,y_train)
```

The results of this meticulous hyperparameter tuning were both significant and promising. We witnessed a tangible improvement in the model's performance, evidenced by a notable reduction in the Root Mean Square Error (RMSE) score, which dropped to 14.2. This decrease in RMSE indicates a model that is more aligned with the complexities and nuances of real-world data and is capable of making predictions with greater accuracy and precision. Moreover, our efforts bore fruit in the form of an increased R-squared (R2) score, which climbed to an impressive 0.79. This improvement in the R2 score signifies that a larger proportion of the variance in our dependent variable is now explainable by the independent variables in our model.

These enhancements in our model's performance metrics are not just numbers; they represent a significant leap in the model's ability to provide reliable, actionable insights. With a more finely tuned model, we are now equipped with a tool that offers a clearer, more accurate window into the dynamics of the job market.

In conclusion, the process of Hyperparameter Tuning and finalizing the Model was a pivotal step in our project. It was a phase where technical rigor met strategic thinking, culminating in a model that stands robust, refined, and ready to make a meaningful impact in the world of job market analytics.



Assessment of our model and process (Did our technique work?)

Our technique, employing data analysis, feature engineering, and machine learning models, **proved effective**. The Random Forest model, in particular, demonstrated robust predictive capabilities after hyperparameter tuning. The project's success lies in its ability to provide practical insights for navigating the complexities of the contemporary job market. Since our main focus is on understanding the competition for jobs, an RMSE of 14.2, which varies on average 14 applications is notable, though more improvements can be done with time and resources.

Conclusion

In conclusion, our "LinkedIn Job Analysis" project has successfully delved into the intricacies of the job market, specifically tailored to assist fellow international students.

At the heart of our project lies a robust predictive model, adept at analyzing job application trends. This model has proven to be an indispensable tool in decoding the complexities of the job market. By predicting the likely number of applicants for various job postings, provides a clear view of the competitive nature of different roles. This insight is invaluable, as it allows job seekers to gauge the level of competition they might face, tailoring their job search strategies accordingly.

Furthermore, our analysis serves as a powerful aid in decision-making for both job seekers and employers. For job seekers, it's like having a compass that guides them through the competitive terrain of job hunting, helping them identify where opportunities lie and what skills are in demand. They can use this information to fine-tune their applications, enhance their skills, and strategically position themselves in areas where they have the best chance of success.

For employers and recruiters, the insights gleaned from our analysis offer a clear understanding of the labor market's supply and demand. It aids them in crafting more effective job descriptions, understanding the pull of various job benefits, and anticipating the number of applicants. This can lead to more efficient recruitment processes and better alignment of job offers with market expectations.

In essence, our project not only illuminates the current state of the job market but also acts as a strategic guide for navigating it. It underscores the importance of data-driven strategies in the modern job search and recruitment processes, highlighting how critical it is to stay informed and adaptable in an ever-changing employment landscape.

Future Improvements

To further enhance the project's impact, we propose the following future improvements:

- **Improved Career Development**: Adding on to our project, we analyzed prevailing job market trends to highlight roles experiencing increased demand. This empowers job seekers with knowledge about which areas are thriving and where they might need to upskill. The goal is to facilitate informed decision-making regarding skill acquisition and certification pursuits, aligned with market needs. This strategic planning could significantly boost employability and job satisfaction, providing a clear, targeted path for professional growth and development.

- **Skill Gap Analysis**: We aim to develop a feature that performs a comprehensive comparison between a user's existing skill set and the qualifications sought in desired job roles. This comparison can pinpoint specific skills the user may need to develop. By providing tailored recommendations for skill enhancement, such as relevant training programs or courses, we address the critical need for continuous learning and adaptation in a fast-evolving job market. This personalized approach to skill enhancement prepares individuals for their aspired roles and ensures their skills remain relevant and competitive. It’s a step towards empowering job

- **Future Predictions**: An additional feature we plan to introduce involves predictive analytics to forecast potential job openings across various sectors. This predictive capability can be a strategic asset for businesses in planning their talent acquisition and workforce management. This foresight into the job market also supports businesses in future-proofing their workforce, ensuring they are well-prepared for emerging industry demands and skill requirements. Likewise, it would support job-seekers to understand where to invest their energy based on the predictions.

These improvements are aimed at maximizing the utility of our project, making it a more actionable tool for both job seekers and employers in navigating the complexities of the modern job market. Our work can also help in implementing additional models for forecasting job openings by industry.

References

- Kaggle. (2023). LinkedIn Job Postings. https://www.kaggle.com/datasets/arshkon/linkedin-job-postings/code