JOB SALARY PREDICTOR

Team Job Consultants:
Akarsh Nagaraj
Mrinal Gupta
Vinith Angadi

Final Project Report
ISE 540 | Fall 2020
December 1st, 2020

GitHub Repo:
https://github.com/akarshn95/Job_Salary_Prediction

Web Application Demo:
https://youtu.be/AKouB8cql3s

01 December 2020

TABLE OF CONTENTS

EXECUTIVE SUMMARY

Our project aims to assist job seekers negotiate salaries by predicting what salary to expect for a certain job description at a given location. There are tens of thousands of jobs on online job boards that invite job seekers to apply for jobs without providing an estimate of the compensation they provide – this was further validated when we were scraping for salaries online. This poses a lot of problems for job seekers, especially new graduates like us, applying for a job role or when trying to negotiate the compensation. These problems are what we seek to solve in our project, where we train machine learning models to predict salaries for a job posting at a certain location in the United States. We, being job seekers, are facing this problem every day in our job hunt which gives us a strong motivation to build an efficient and reliable model to help us in finding the right compensation for our jobs.

In order to accurately predict salaries for a job posting, we extracted the best features from the job postings scraped off the internet and trained our models using these as inputs. With a verbose job description having most of the important predictors, the text was cleaned and preprocessed by tokenizing and vectorization. We also implemented TF-IDF vectorization for the model to understand the important words in a document. We recognized that the salary, which is our target, depended largely on the title of the job posting. We grouped similar jobs in the same field together by implementing topic modeling with non-negative matrix factorization. By manually annotating the different topics or groups, we were able to add a generic job title as a predictor to our model. The features extracted from the job description, the job location, the topics obtained by topic modeling along with cost index and purchasing power parity of each state helped us to train accurate models to predict job salary. The idea behind adding cost index and purchasing power parity was to include the information of the different pay scales in different states for a similar job title. For example, salary offered to a Data Scientist in California would be way higher than the same offered in Nevada because of the difference in the standard of living, hence, we wanted the model to learn this behaviour.

We initially trained linear regression, ridge regression and lasso regression models to predict the salaries. Linear models couldn't truly capture the relationship between the target and predictors, which prompted us to implement tree-based models like Random Forest regressor and Gradient Boosted Trees. After using GridSearchCV for hyperparameter tuning, we were able to get encouraging results from the models, the best of which (Gradient Boosting) predicted the target salary with a mean absolute error of $9,112. To enable job seekers to effortlessly make use of our salary prediction model, we used Flask API to host our model on a server with an easy-to-use UI. The resultant web application takes in job description and location from the user and outputs the estimated salary for the job posting. While processing the data and building our models, we were able to draw some interesting insights from the job posting data which we have showcased in the report.

PROPOSED APPROACH

DATA COLLECTION

In order to predict the salary from the set of predictors, we needed to collect a lot of data to train the machine learning models. We wanted to work with raw data from the real-world, hence we chose to scrape data off Indeed.com, a popular job board with variety of jobs. We used Selenium web scraper with search strings ranging from 'Software Engineer', 'Data Scientist', 'Project Manager', 'Mechanical Engineer', 'Accountant' to 'Cashier' to scrape 30,000 job postings. The final scraped data was a CSV file containing the job listings with job descriptions and location.

DATA CLEANING & PREPROCESSING

The meat of information about any job listing is present in its job description which prompted us to thoroughly clean, process and extract features from the job description column. The salary column too was highly inconsistent with different formats of salary like hourly, monthly and yearly which had to be standardized. The cleaning was performed using regular expressions and stopwords from NLTK library.

- Job Description column cleaned by removing punctuation, new line characters, extra spaces/tabs and stopwords.
- Salary column standardized by converting all formats to salary (in $) per year using RegEx and conversion rules.
- States extracted from job location column and standardized to remove spelling errors. The fig.1 below shows a sample of an unclean data (left) and a clean data (right):

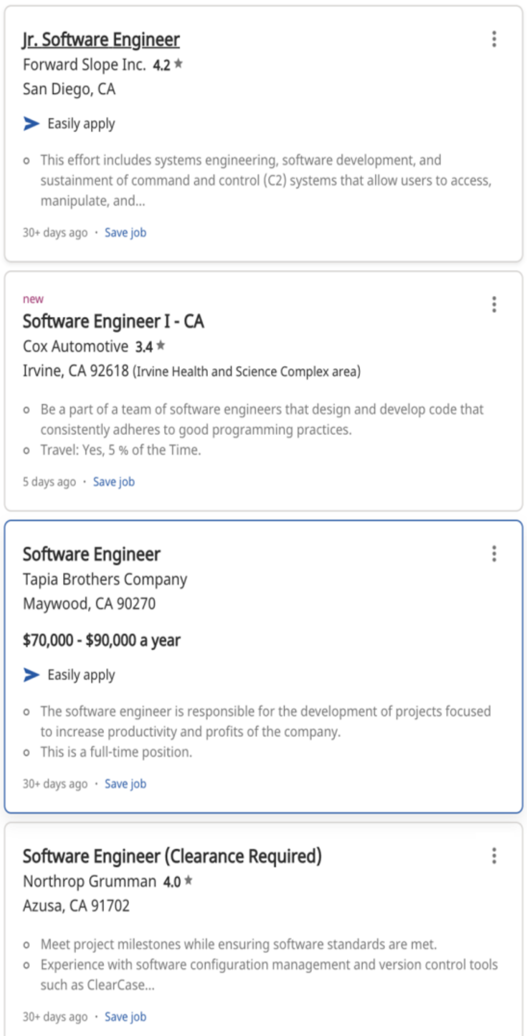
46−49 an hour	Goleta, CA 93117
9, 588−13,460 a month	Fresno, CA 93721 (Central area)
$8,000 a month	Industry, CA 91789
From $17 an hour	None
90, 000−95,000 a year	Los Angeles, CA 90028 (Hollywood area)
32−50 an hour	Boston, MA
135, 000−185,000 a year	None
75, 000−100,000 a year	None

salary_offered	state
62500.0	VA
62000.0	MA
156000.0	MA
79040.0	CA
65395.0	TN
65395.0	TN
62500.0	CO
52000.0	TX

Figure 1. Unclean Salary and State features (left) and clean Salary and State features (right)

- With a pre-defined set of vocabulary of 7000 skills (found on internet) which contained a mix of technical as well as non-technical skills, we converted the job description text to a Term Frequency – Inverse Document Frequency (TF-IDF) vector by extracting only the matching skills from that vocabulary.
- Added cost index and purchasing power parity of each state for the job listings.

The final data after scraping, cleaning and preprocessing before vectorization looks like Fig. 2 below:



Title	Job_Description	Job_Type	States	Salaries
Mechanical Engineer	job description mechanical engineer comp	fulltime	FL	57500
Mechanical Engineer	description the mechanical engineer inter	fulltime	MI	78000
Licensed Mechanical Design	looking for a licensed p e with 5 years expe	fulltime	ID	76000
Mechanical Engineer	in this role you will use engineering method	fulltime	MO	70000
Senior Engineer - Suspensior	position senior engineer suspension desig	fulltime	CA	115000
Mechanical Engineer	mechanical engineer job description respo	fulltime	WA	70000
Mechanical Engineer (SHUKI	job description mechanical engineer comp	fulltime	FL	57500
Electro-Mechanical Support	electro mechanical support engineer you v	fulltime	MA	72000
Administrative Specialist (M	under minimal supervision this position is r	fulltime	FL	39500
Quality Engineer	ortho development is a fast growing compa	fulltime	UT	72565.5
Contract Mechanical Design	this is a 6 month contract opportunity with	fulltime	MA	65520
Manufacturing Engineer	we have a full time direct hire opportunity	fulltime	MO	70000
General Engineer	duties summary these positions are loca	fulltime	MA	108053.5
Supervisory General Enginee	duties summary this position is located in	fulltime	VA	105423.5
Manufacturing Process Engi	1 0 about ess our team and what we are l	fulltime	OR	85000
Interdisciplinary	duties summary about the position this is	fulltime	TX	101448.5
Manufacturing and Controls	intigral is the fenestration industry s fully in	fulltime	OH	75000
Sr Product Manufacturing E	due to contractual requirements us citizen	fulltime	FL	83200
Mechanical Engineer	oceangate inc is a leader in technologically a	fulltime	WA	80000
Mechanical Engineer - Biling	job responsibilities and duties 1 generate c	fulltime	CA	52000
Mechanical Engineer	manufacturing mechanical engineer entry	fulltime	PA	64000
Mechanical Engineer	amentum is looking for a mechanical engin	fulltime	VA	138000
Mechanical Designer	mechanical designer linwood engineering i	fulltime	CA	74000
Mechanical Engineer - Interr	the mechanical engineer intermediate will	fulltime	MI	67600
INTERDISCIPLINARY ENGINE	duties summary you will serve as the facili	fulltime	HI	101805.5
Mechanical Engineer(Chines	new england only generate cad models and	internship	MA	49920
Quality Engineer	job title quality engineer department engi	fulltime	CA	70000
Plastic Injection Mold Proce	looking for a highly motivated self starter to	fulltime	CO	59280
Senior Mechanical Engineer	senior mechanical engineer p e atlantic de	fulltime	MD	92500

Figure 2. Indeed.com job cards (left) and scraped and cleaned data (right)

Topic Modeling

The domain of the job is a very important factor in predicting the salary, since the domain tells us the required skills and these skills usually directly influence the resultant salary. This posed an unsupervised learning problem to us since the job domain was not labeled and we needed to group and identify the job field from the job description. We implemented topic modeling using non-negative matrix factorization where each job listing was grouped into one of the 10 topics. We manually annotated and provided labels to these 10 topics by looking at the top words in each of the topic. The 10 topics generated were namely Data Science, Software Engineering, Data Analytics, Mechanical Engineering, Engineering Manager, Manager, Cashier, Accounting, Customer Service, and Sales and Marketing. These topics were added to the data as one of the predictors for every job listing. Two topics with their top 30 words and annotation are shown below in Figure 3:

```
Topic #2: (Software Engineering)
software,database,agile,technology,sql,cloud,software
development,etl,integration,computer,intelligence,architecture,infrastruct
ure,computer science,programming,business intelligence,big data,data
engineering,scrum,warehouse,management,data warehouse,automation,business
requirements,security,data management,data
processing,implementation,linux,innovation

Topic #3: (Mechanical Engineering)
manufacturing,product development,mechanical
engineering,solidworks,communication,assembly,cross-functional,root,cad,gm
p,analytical,technology,automotive,manufacturing engineering,written
communication,troubleshooting,problem solving,chemical engineering,lean
manufacturing,process
improvement,curriculum,interpersonal,solid,mathematical,automation,plc,hea
t,technical support,inspection,supervision
```

Figure 3. Topic Modelling results

The Model

After the cleaning, preprocessing, and topic modeling steps, our data had job location, cost index, purchasing power parity, TF-IDF vector of job description and the job domain obtained from topic modeling. With one-hot encoding of the categorical features and TF-IDF vector, the final 'X' vector of all the predictors consisted of 7241 columns. The 'y' or target of our prediction was the salary for a job posting – making it a regression problem. We used StandardScaler to normalize cost index, purchasing power parity and the TF-IDF vector before training our models.

We initially tried a Linear Regression model, but the model was too 'simple', and a straight line couldn't capture the relationship between X and y. Also, due to multicollinearity induced from the TF-IDF vector, the Linear Regression model had large errors. In order to take care of multicollinearity and regularize the model, we implemented L2 regularization with Ridge Regression. This gave us a much-improved linear model with closer predictions of the target salary.

However, since linear models assume the presence of a linear relationships between the predictors and target, they couldn't provide the best of results. We needed a more 'complex' model capable of capturing such a relation. Random Forests and Gradient Boosted Trees are a great alternative for these kinds of problem since they learn freely based off of the data without any assumed relationship between the predictor and target variables. Also, these tree-based models avoid overfitting by aggregating results (in case of Random Forests) and incremental learning (in case of Gradient Boosted Trees), which make them good models for our regression problem.

FLASK API

To enable ease of access and use of our project, we hosted it on a server with Flask API. The user is prompted to enter the job description and location and the predicted salary is displayed on the webpage. The model communicates with the user via the Flask API.

Flask Web Application UI

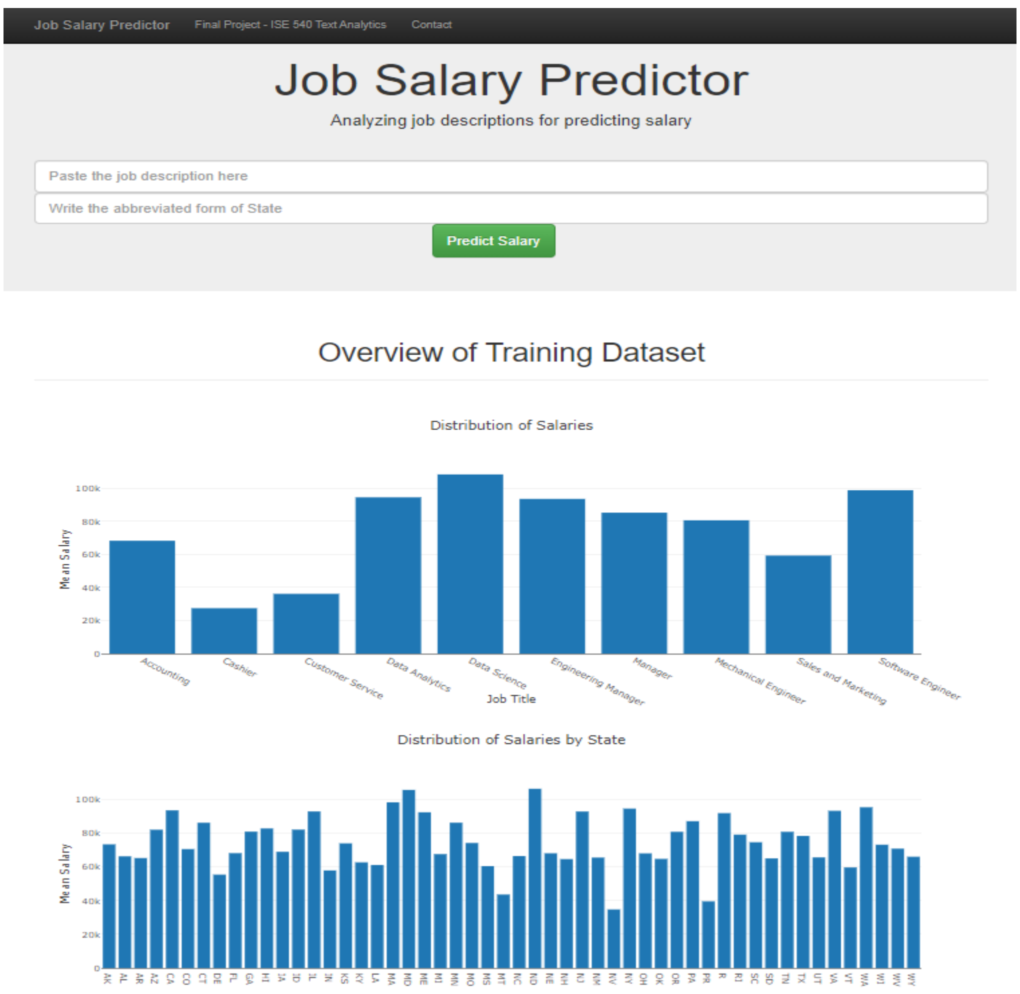


Figure 4. Web Application Home Page showing some training data insights

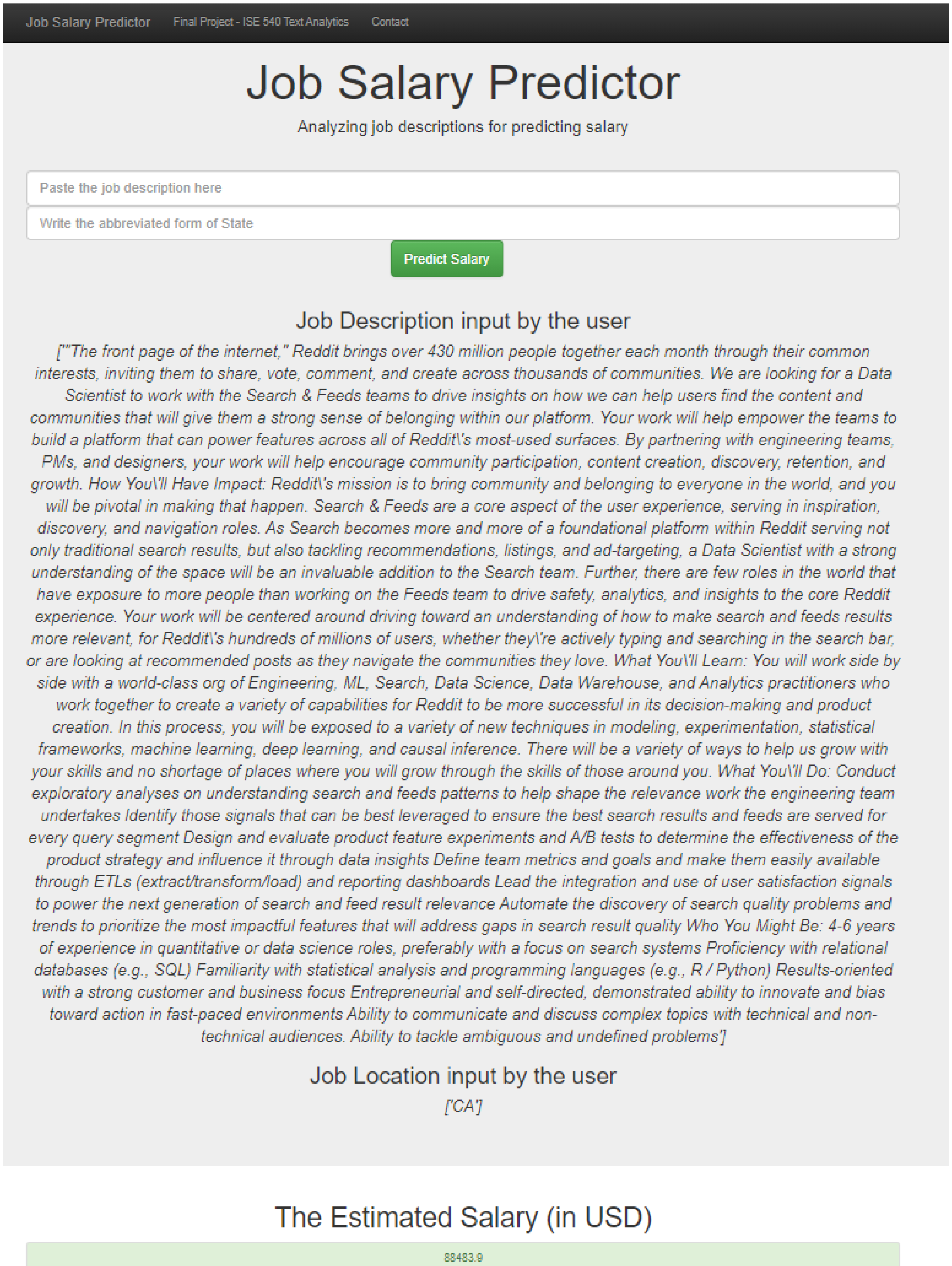


Figure 5. Prediction Result

EXPERIMENTAL RESULTS

ERROR METRICS

Since our problem is a regression problem predicting a continuous variable salary as the target, we used the following standard metrics for evaluating regression models:

- **MEAN ABSOLUTE ERROR (MAE)** – measures the average magnitude of the errors in a set of predictions, without considering their direction. Lower, the better.

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n}$$

MAE = mean absolute error
y_i = prediction
x_i = true value
n = total number of data points

- **MEAN ABSOLUTE PERCENTAGE ERROR (MAPE)** – the average of the absolute percentage errors of predictions. Error is defined as actual or observed value minus the forecasted value. Percentage errors are summed without regard to sign. Lower, the better.

$$M = \frac{1}{n} \sum_{t=1}^{n} \left| \frac{A_t - F_t}{A_t} \right|$$

M = mean absolute percentage error
n = number of times the summation iteration happens
A_t = actual value
F_t = forecast value

- **CO-EFFICIENT OF DETERMINATION (R2)** – the proportion of the variance in the dependent variable that is predictable from the independent variables. It is the goodness of fit of the regression model. Higher, the better.

$$R^2 = 1 - \frac{RSS}{TSS}$$

R^2 = coefficient of determination
RSS = sum of squares of residuals
TSS = total sum of squares

Model Training

We used 4-fold cross validation with 80% train and 20% test split for all metric evaluations. We calculated some baseline metrics to get an understanding of how much better our models were performing. The baseline MAE is simply the mean salary from our dataset, the baseline MAPE is the MAPE by assuming the mean salary is the predicted value for every observation and the baseline R^2 is the co-efficient of determination of line *y = mean salary*. The baseline metrics for our salary prediction problem are as follows:

- Baseline MAE: **$28,150**
- Baseline MAPE: **54.6%**
- Baseline R^2: **0**

The following models were trained and cross validated to arrive at the below metrics. These are the results from standard models without any hyperparameter tuning.

Models Tested	MAE Without TF-IDF (in $)	MAPE without TFIDF	MAE with TF-IDF (in $)	MAPE with TF-IDF
Linear Regression	22264	37%	8.47×10^{17}	11.39×10^{12}
Ridge Regression	22257.72	37.06%	26722.16	39.8%
Lasso Regression	22264.3	37%	28921.12	44.1%
Random Forest	21925.68	37%	16492.05	25.9%
Gradient Boosting	21863	36.35%	20212.42	33.46%

Table 1. Test Results without hyperparameter tuning

Linear Regression – It performed better than the baseline, but due to multicollinearity in TF-IDF vector, the model performs worse than the baseline and that is why you can see a huge variance in the error results.

Ridge and Lasso Regression – They both performed much better than the baseline and the multicollinearity arising from the TF-IDF vector is taken care of by the regularization co-efficient.

Random Forest – Random forest with TF-IDF performed the best among all the models. The aggregated learning approach by building multiple decision trees with varied ‘weaknesses’ are compensated when taking the aggregated result. This also helps in avoiding overfitting the data.

Gradient Boosting – This model with its incremental learning approach gives good predictions which are much better than the baseline. Coupled with TF-IDF, this is the second-best performing model with nearly one-third percentage error.

Hyperparameter Tuning

We performed hyperparameter tuning on Ridge Regressor, Random Forest regressor and Gradient Boosting regressor thereby improving their overall performance. The following are the results of the best parameters obtained from hyperparameter tuning with GridSearchCV (4-fold).

Model	Best Parameters	MAE (in $)	MAPE	R^2
Ridge Regressor	{'alpha': 1.0}	27,586	41.40%	94.06%
Random Forest	{'max_depth': 60, 'max_features': 1400, 'n_estimators': 450}	11,453	17.02%	95.11%
Gradient Boosting	{'learning_rate': 0.1, 'max_depth': 10, 'n_estimators': 1000}	9,112	13.37%	99.65%

Table 2. Hyperparameter Tuning Results

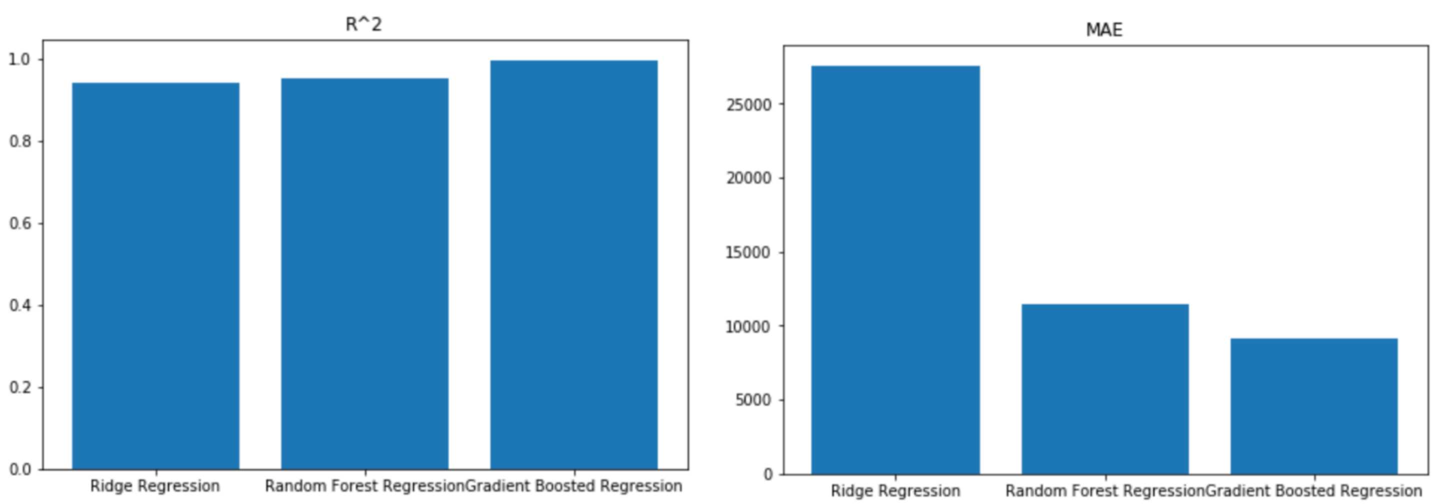


Figure 6. R^2 and MAE values for three models after hyperparameter tuning

Statistical Significance Testing

To make tangible conclusions about our models' performance with respect to the baseline, we must be certain that the results are statistically significant and not a result of randomness. Hence, we conducted a paired t-test with 95% confidence between absolute error of baseline and each of the 3 models namely Ridge, Random Forest, and Gradient Boosting with the following null and alternative hypothesis for each of the 3 models.

H_0: mean absolute error of baseline and the model is the same
H_a: mean absolute error of baseline and the model is different

Model	t-statistic	p-value	Null Hypothesis	Inference
Ridge Regression	-0.263	0.792	Accept (p-value too large)	Ridge and baseline are not significantly different from each other
Random Forest	-16.374	1.2×10^{-55}	Reject (p-value is very small)	MAE of model is significantly lower than that of baseline as indicated by negative t-stat
Gradient Boosting	-17.9554	1.54×10^{-65}	Reject (p-value is very small)	MAE of model is significantly lower than that of baseline as indicated by negative t-stat

Table 3. Statistical Significance Testing Results

DISCUSSION

- After statistical significance testing of the experimental results and the large magnitude of t-statistic, we can safely say that the Random Forest regressor and Gradient Boosting regressor performed much better than the baseline model. By implementing topic modeling, TF-IDF and data preprocessing, we were able to successfully bring down the MAE from over $28,000 for the baseline to about $9,000 in case of Gradient Boosting regressor model.
- The breakdown of mean salaries per each domain identified by topic modeling is as shown below in Fig. 7. We can see the clear distinction in the salaries as one would expect in the real-world and our topic modeling algorithm was able to capture it well.

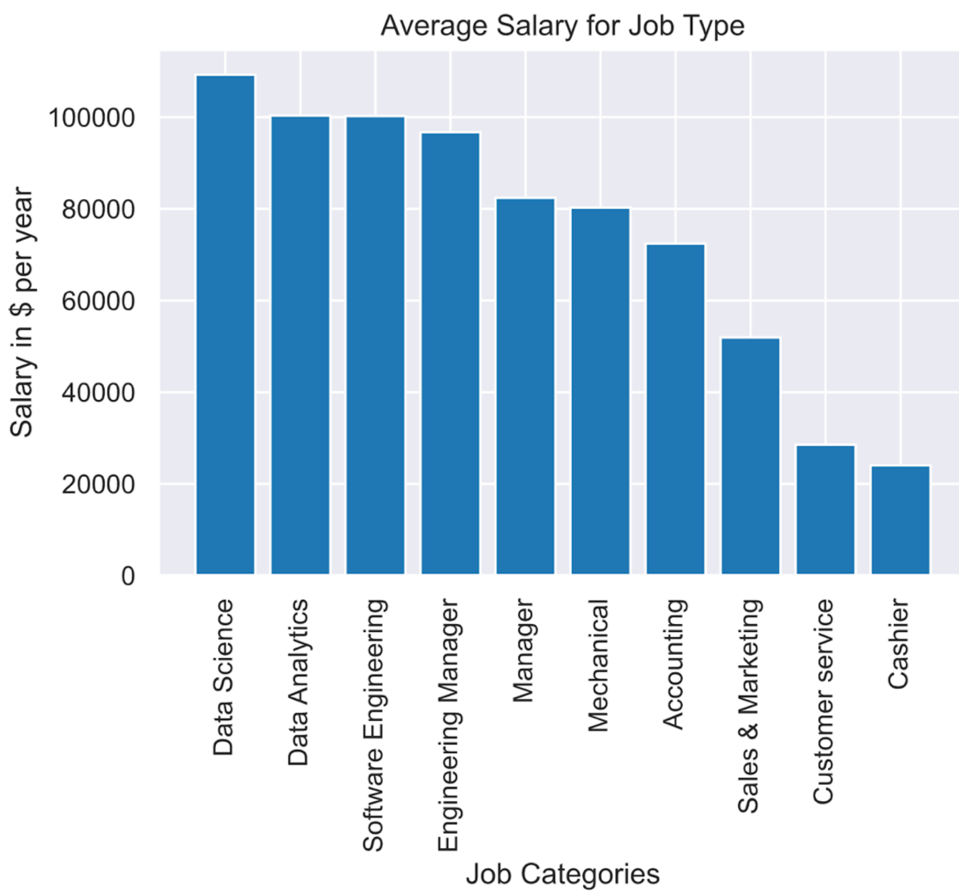


Figure 7. Mean Salaries for each topic

- We uncovered some interesting insights when working with the real-world job postings dataset. While building the model, we were able to identify which factors heavily influence the compensation provided (positively and negatively). The following figure shows the most important features generated from the Random Forest model:

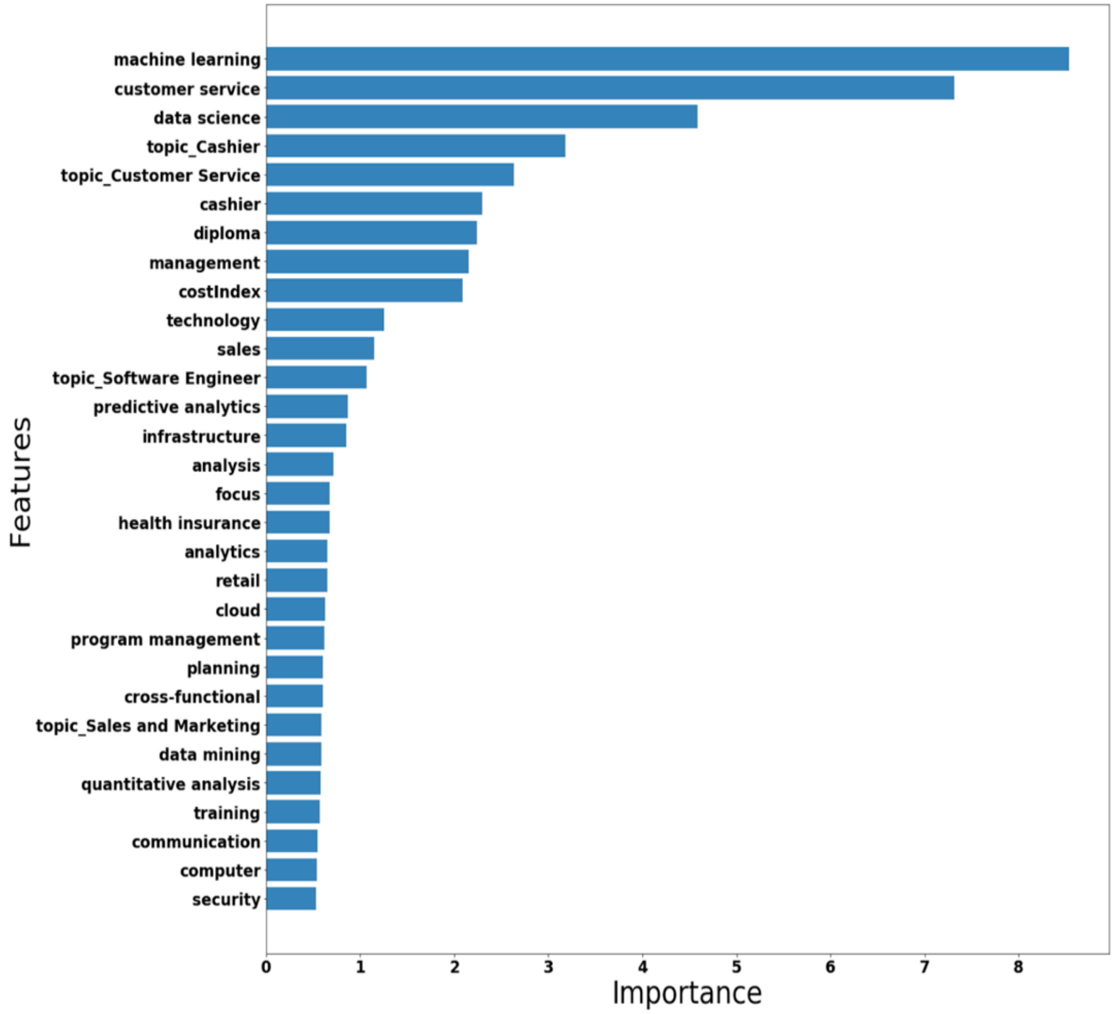


Figure 8. Top 30 Features from Random Forest

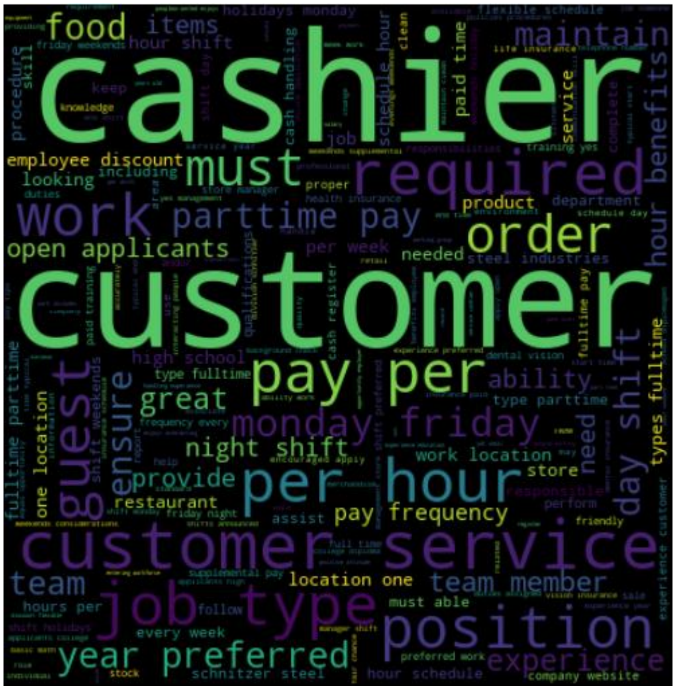
Qualitative Analysis of a few features

Machine learning: This feature is an example where the skill has a large positive influence on the resultant salary.

Topic_Cashier: This feature is an example where the job domain has a significant negative influence on the resultant salary as the mean salary for the topic_cashier was least among all the other topics. This correlates with the real-world where the cashiers are paid lesser than the other job titles

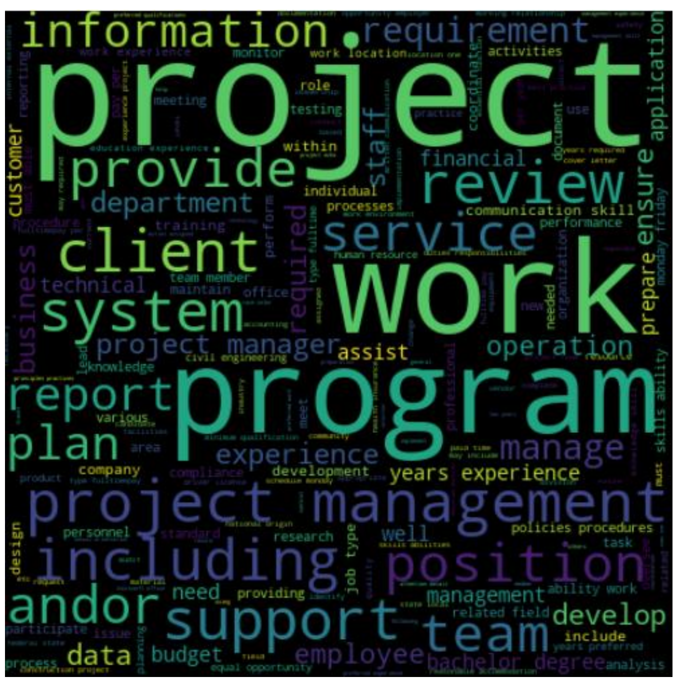
CostIndex: We can see that the costIndex corresponding to a state has a large impact on what the salary is going to be and this shows that as the state changes, our salary results also changes which shows that we are successful in injecting that information in our model.

- From the results of topic modeling of the job descriptions, we were able to clearly recognize the most important skills in demand for each domain. These insights, arising from real-world live data can help job seekers understand the skill requirements of employers. This in-turn can help them be best prepared when trying to break into the industry. The following word clouds shows the most important skills that are required for Data science, Cashier, and Manager.



Cashier

Data Science



Manager

CONCLUSION

Our best model, Gradient Boosting regressor after hyperparameter tuning, was able to predict the salary from a job description and location with a Mean Absolute Error (MAE) of $9,112 and Mean Absolute Percentage Error (MAPE) of 13%. This is a significant improvement from the baseline model which had MAE of $28,150 and MAPE of 54.6%. Aggregated models performed much better than simple linear models in our regression problem since there wasn't a simple linear relationship between the predictors and the target variable. Our web application provides job seekers the ease of getting an expected salary from a job description and location.

Future Scope

- Our model is limited by the data; having more data with the salary column that can be used to train the model will enable better predictions.
- Inclusion of more predictor variables of a job posting such as company size, company rating, year established, change in stock price etc. will help improve the model performance.
- Other NLP techniques specifically Word2Vec can be used to understand the semantic meanings of the words and learn word associations in the job description.

Lessons Learned

- Web scraping is hard; so is data cleaning because of inconsistency in real-world data
- Data Collection & Data Cleaning constitutes a significant part of a Data Science project.

APPENDIX

The complete results of the 10 topics generated and the top 30 words of each topic is shown below:

Data Science:
machine learning, data science,algorithms,computer,python,modeling,big data,programming,phd,technology,r,cloud,impact,communication,artificial intelligence,software,natural,innovation,analytics,statistics,computer science,focus,mathematical,leadership,intelligence,communication skills,optimization,open source,cross-functional,visualization

Manager:
management,project management,compliance,planning,communication,budget,reporting,leadership,implementation,organizational,training,communication skills,interpersonal,monitoring,contracts,analysis,written communication,documentation,supervision,focus,microsoft office,budgets,controls,software,compensation,computer,quality control,time management,analytical,audit

Sales & Marketing:
sales,retail,marketing,automotive,training,compensation,business-to-business,communication skills, management, focus,organized,exceed,communication,prospecting,crm,business-to-business sales,interpersonal,customer experience, commitment, wireless,motivation,pricing,positive attitude,inventory,customer focus,hospitality,pos,strategy,outside sales,organizational skills

Engineering Manager:
security,security clearance, documentation,training,dod,act, management,curriculum,recruitment,human resources,claims,physics,pass,calculus,heat,requirements engineering,supervision,opm,investigation,evaluation,technology,planning,acquisition,provision,performance appraisal,program management,civil engineering,mathematical,computer,analysis

Customer service:
customerservice,retail,diploma,safe,training,fit,payments,ged,communication,communicationskills,receiving,interpersonal,csr,computer,supervision,time management,quality control,clerk,pass,expect,customer satisfaction,san,tip,layout,cash handling,doe,sorting,loss prevention,purchasing,organizational

Mechanical:
manufacturing,product development,mechanical engineering, solidworks,communication,assembly,cross-functional,root,cad,gmp,analytical,technology,automotive,manufacturing engineering,written communication,troubleshooting,problem solving,chemical engineering,lean manufacturing,process improvement, curriculum, interpersonal,solid,mathematical,automation,plc,heat,technical support,inspection,supervision

Data Analytics:
analytics,analysis,analytical,data analysis,marketing,modeling,sql,visualization,reporting,advanced analytics,statistics,strategy,impact,optimization,data visualization,computer,data mining,programming,r,tableau,metrics,database,intelligence,sas,communication,mathematics,predictive analytics,leadership,cross-functional,presentations

Accounting:
health insurance,life insurance,payroll,professional development,finance,accuracy,civil engineering,fit,spanish,accounts payable,cpa,general ledger,autocad,communication,microsoft office,bookkeeping,quickbooks,financial statements,software,transportation,pass,audit,computer,accounting principles,communication skills,human resources,compliance,payments,focus,organized

Software Engineering:
software,database,agile,technology,sql,cloud,software development,etl,integration,computer,intelligence,architecture,infrastructure,computer science,programming,business intelligence,big data,data engineering,scrum,warehouse,management,data warehouse,automation,business requirements,security,data management,data processing,implementation,linux,innovation

Cashier:
cashier,diploma,organized,cash handling,fit,safe,payments,focus,accuracy,creativity,receiving,positive attitude,pos,inventory,swing,mathematical,retail,communication,natural,reach,automotive,5s,pricing,sorting,retail operations,layout, bilingual,dexterity,magic,supervision