REAL / FAKE JOB POSTING PREDICTION

INFORMATION AND NETWORK SECURITY

DINESH MARUPUDI VEERA – 999901286
VIVEK KIRAN KATARAM – 999902375
UDITHA REDDYKATTEGUMMULA - 999902201

Abstract:

Employment fraud is becoming more prevalent. According to CNBC, there were twice as many employment frauds in 2018 than there were in 2017. The state of the market today has resulted in substantial unemployment. Numerous people have experienced much less job loss and economic stress because of the coronavirus. Such a situation offers con artists the ideal opening. Due to a rare incidence, many people are becoming victims of scammers who feed on their despair. Most con artists use this technique to obtain personal information from their victims. Addresses, bank account information, social security numbers, and other personal information are examples. As a student, I have encountered several of these fraudulent emails. The con artists offer their victims incredibly lucrative career opportunities and then demand payment in exchange. Or they demand money from the job seeker in exchange for the promise of employment. Natural Language Processing and machine learning approaches can be used to solve this severe dilemma (NLP).

This project makes use of data from Kaggle. The characteristics of a job ad are contained in this data. These job listings are either labeled as genuine or bogus. A very small portion of this collection consists of fake job listings. That is acceptable. We don't anticipate seeing many fake job postings. This project is divided into five stages. The five steps of developing our machine learning model include:

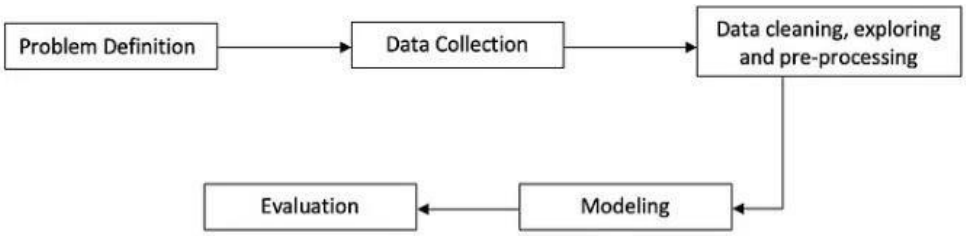


Figure 1: Fake jobs machine learning stages

Research Problem:

The goal of this project is to develop a classifier that can distinguish between phony and legitimate jobs. Two distinct models are used to evaluate the outcome. One model will be used on the text data and another on the numeric data since the submitted data comprises both text and numeric properties. The result will be a synthesis of the two. The final model will incorporate all pertinent information from job postings and generate a conclusion on whether the position is legitimate or not.

1. What are the most important features in predicting whether a job posting is real or fake?
2. How accurate are different machine learning algorithms in predicting whether a job posting is real or fake?
3. Are there any significant differences in the characteristics of real job postings compared to fake job postings?
4. Can natural language processing techniques improve the accuracy of predicting whether a job posting is real or fake?
5. Is there a correlation between the geographical location of the job posting and its likelihood of being fake?

Introduction:

The development of the internet has significantly streamlined the hiring process. Additionally, the ongoing pandemic has significantly contributed to the present shift in the pattern of job recruitment. Online hiring has made it possible to find more prospects and streamline the hiring process, and it has been very helpful in bridging the gap between recruiters and potential candidates. With only the press of a mouse, candidates may now apply online to a wide number of positions based on their area of expertise. E-recruitment assists businesses in utilizing a variety of internet-based options. Users can broaden their employment searches and find the best applicants by using online recruitment where they can converse with competent prospects from around the world. When a client relies on internet recruitment, the best candidate is ultimately hired. Candidates' online personas can be discovered through social media sites like Facebook and LinkedIn. Companies can choose competent individuals and increase efficiency by using tools like pre-employment screening, personality assessments, and tests for candidate screening.

The process of online recruitment involves minimal human interaction. Communication costs are lower, making it more cost-effective. However, some advertised job openings are fraudulent and serve as bait to obtain personal data, instead of genuine job opportunities. When candidates apply for these jobs, their potential information is stolen, or hackers gain access to their computers to steal important data. Cybercriminals may combine victim data and sell it on the dark web or continue to use it for years. Research has been conducted on detecting fake job postings using machine learning algorithms and ensemble classification modelling to enhance accuracy. By utilizing an appropriate dataset, sufficient analysis, cleaning, and machine learning techniques, the identification of false job postings is possible.

Dataset exploration Metrics:

The models are evaluated based on two metrics:

Accuracy: This formula defines this metric –

$$Accuracy = \frac{True\ Positve + True\ Negative}{True\ Positive + False\ Positive + False\ Negative + True\ Negative}$$

As the formula suggests, this metric produces a ratio of all correctly categorized data points to all data points. This is particularly useful since we are trying to identify both real and fake jobs, unlike a scenario where only one category is essential. There is, however, one drawback to this metric. Machine learning algorithms tend to favor dominant classes. Since our classes are highly unbalanced, a high accuracy would represent how well our model categorizes the negative category (real jobs)

$$F_1 = \frac{True\ Positve}{True\ Positive + \frac{1}{2}\ (False\ Positive + False\ Negative)}$$

F1-score is used because, in this scenario, both false negatives and false positives are crucial. This model needs to identify both categories with the highest possible score since both have high costs.

https://www.kaggle.com/shivamb/real-or-fake-fake-jobposting-prediction has the data for this experiment. 17,880 observations and 18 characteristics make up the dataset. The information consists of three different data types: integer, boolean and text. Below is a quick definition of the variables:

#	Variable	Datatype	Description
1	job_id	int	Identification number given to each job posting
2	title	text	A name that describes the position or job
3	location	text	Information about where the job is located
4	department	text	Information about the department this job is offered by
5	salary_range	text	Expected salary range
6	company_profile	text	Information about the company
7	description	text	A brief description about the position offered
8	requirements	text	Pre-requisites to qualify for the job
9	benefits	text	Benefits provided by the job
10	telecommuting	boolean	Is work from home or remote work allowed
11	has_company_logo	boolean	Does the job posting have a company logo
12	has_questions	boolean	Does the job posting have any questions

Fig-2: Model Feature Description

A summary statistic is not required in this case because most of the datatypes are either booleans or text. Job_id is the lone integer, which is irrelevant for our investigation. To find null values, the dataset is further examined.

```
Out[5]: job_id                   0
        title                    0
        location               346
        department           11547
        salary_range         15012
        company_profile       3308
        description              1
        requirements          2695
        benefits              7210
        telecommuting            0
        has_company_logo         0
        has_questions            0
        employment_type       3471
        required_experience   7050
        required_education    8105
        industry              4903
        function              6455
        fraudulent               0
        dtype: int64
```

Figure 2: The null value dataset count

The values for many variables, including department and salary range, are missing. The remaining analysis skips these columns.

A first analysis of the dataset revealed that the job posts were in various languages because ethey were taken from various nations. Nearly 60% of the dataset used for this research comes from US-based sources. All the data is in English for simple interpretation thanks to the data from US-based locales. For additional study, the location is divided into the state and the city. The final dataset has 20 features and 10593 observations.

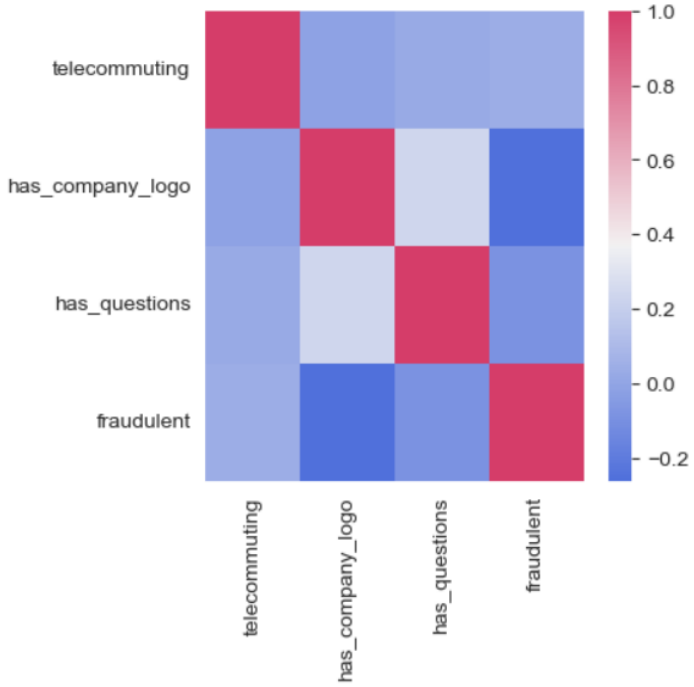


Figure 3: dataset correlation matrix

With only 725 or 7% of the jobs in the sample being fraudulent, only 9868 or 93% of the positions in the dataset are actual. a chart of counts can very clearly demonstrate the discrepancy.

Sample Code- Processing:

Below is the list of sample data pre-processing and processing steps that were followed.

1. Screenshot below shows the removal of undesired columns from dataset and clearing out the null and nan values in text and categories columns. The last step shows the result of total real vs fake jobs – which is about 5% of total postings.

```
In [7]:
#Removing undesired columns & nan
data.function.fillna(data.department,inplace=True)
data.drop(columns=['job_id','salary_range','department'],inplace=True)
```

```
In [8]:
#Now handling the missing values for text data and categorical data

text=['title','benefits','company_profile','location','description','requirements','fraudulent']

categ=['employment_type','required_experience','required_education','industry','function','telecommuting','has_company_logo',

#filling nan in categorical data
categ_cols=data[categ].fillna('None')

#filling nan in text data
txt_cols=data[text].fillna(' ')


categ_cols['country']=txt_cols['location'].apply(lambda x:x.split(',')[0])
countries=categ_cols['country'].value_counts().to_frame()
```

```
In [23]:
#toal of Real and Fake Jobs in the dataset

data['fraudulent'].value_counts().to_frame()
```

Out[23]:

	fraudulent
0	17014
1	866

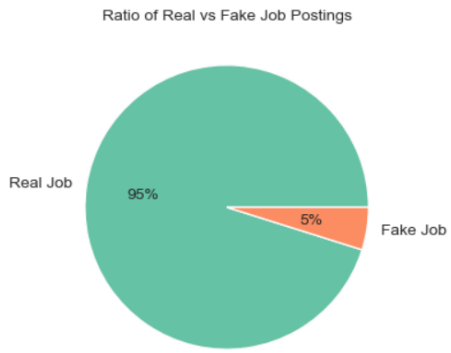
2. Screenshot below shows the pie-chart plotting for the said ratio.

```
In [24]:
#Plotting the ratio of in pie-chart

colors = sns.color_palette('Set2')[0:10]
labels=['Real Job','Fake Job']
plt.figure(figsize=(6,4))
plt.title('Ratio of Real vs Fake Job Postings',size=10)
sns.set_style('whitegrid')

#plotting pie-chart
plt.pie(data['fraudulent'].value_counts(),labels=labels,colors=colors,autopct='%.0f%%')
```

```
Out[24]: ([<matplotlib.patches.Wedge at 0x2c8ce183730>,
  <matplotlib.patches.Wedge at 0x2c8ce191070>],
 [Text(-1.0872905906487755, 0.16673083544034975, 'Real Job'),
  Text(1.0872905828435406, -0.16673088634009564, 'Fake Job')],
 [Text(-0.593067594899332, 0.09094409205837257, '95%'),
  Text(0.5930675906419312, -0.09094411982187034, '5%')])
```



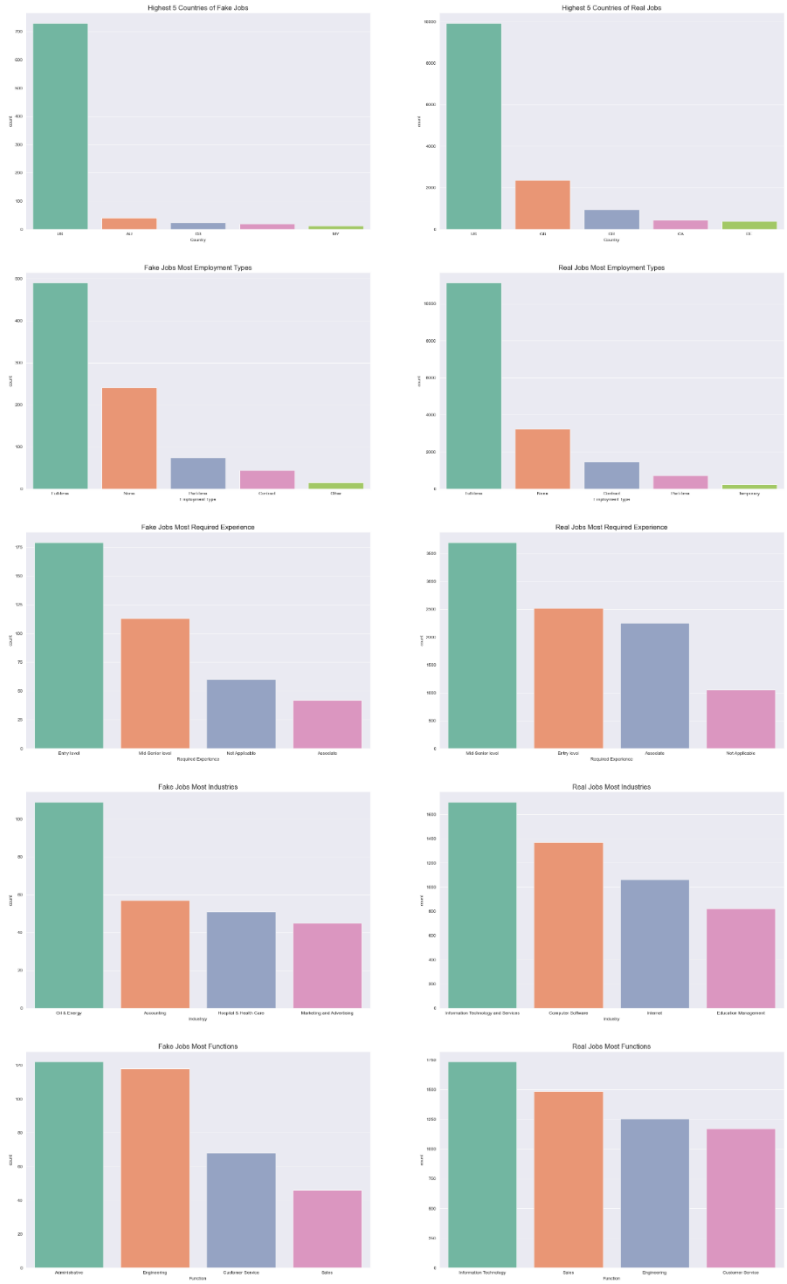
3. Screenshot below shows a part of the step where the difference of real and fake postings in various scenarios are defined and plotted accordingly.

```
In [11]:
fakejobs=categ_cols[categ_cols['fraudulent']==1]
realjobs=categ_cols[categ_cols['fraudulent']==0]
sns.set_style('darkgrid')
fig,axes=plt.subplots(5,2,figsize=(30,50))

# Highest 5 Countries of Real/Fake Jobs
sns.countplot(fakejobs['country'],palette='Set2',order = fakejobs['country'].value_counts()[:5].index,ax=axes[0,0])
sns.countplot(realjobs['country'],palette='Set2',order = realjobs['country'].value_counts()[:5].index,ax=axes[0,1])
axes[0,0].set_title('Highest 5 Countries of Fake Jobs',fontsize=15)
axes[0,1].set_title('Highest 5 Countries of Real Jobs',fontsize=15)
axes[0,0].set_xlabel('Country')
axes[0,1].set_xlabel('Country')

# Real/Fake Jobs Most Employment Types
```

4. Screenshot below shows various scenarios of plotting between real and fake jobs.



5. Screenshot below shows the plotting of top 3 list of categories.

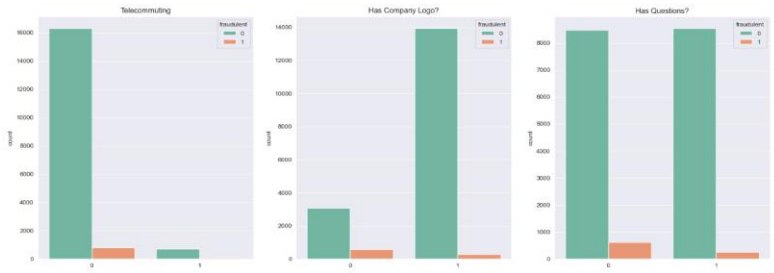
```
In [12]: #Plotting list of jobs based on top list of categories

         fig,axes=plt.subplots(1,3,figsize=(21,7))
         sns.countplot(x='telecommuting', data=categ_cols, hue='fraudulent', palette='Set2',ax=axes[0])
         sns.countplot(x='has_company_logo', data=categ_cols, hue='fraudulent', palette='Set2',ax=axes[1])
         sns.countplot(x='has_questions', data=categ_cols, hue='fraudulent', palette='Set2',ax=axes[2])

         axes[0].set_title('Telecommuting',fontsize=12)
         axes[1].set_title('Has Company Logo?',fontsize=12)
         axes[2].set_title('Has Questions?',fontsize=12)

         axes[0].set_xlabel('')
         axes[1].set_xlabel('')
         axes[2].set_xlabel('')

Out[12]: Text(0.5, 0, '')
```



Future Work: Algorithm and Techniques

It is clear from the preliminary study that text and numerical data must both be employed for the final modeling. A final dataset is selected prior to data modeling. For the final analysis, this project will employ a dataset with the following characteristics:

1. Telecommuting
2. Fraudulent
3. ratio: fake to real job ratio based on location
4. text: a combination of title, location, company_profile, description, requirements, benefits, required_experience, required_education, industry, and function
5. character_count: Count of words in the textual data Word count histogram
6. Further pre-processing is required before textual data is used for any data modeling.

The algorithms and techniques that we expected to use in the project are:

- Naïve Bayes Algorithm
- SGD Classifier
- Natural Language Processing

References:

1. Ahmed, H., Traore, I., & Saad, S. (2017). Detection of online fake news using n-gram analysis and machine learning techniques. In International conference on intelligent, secure, and dependable systems in distributed and cloud environments, 127-138.
2. Allcott, H., & Gentzkow, M. (2017). Social media and fake news in the 2016 election. Journal of economic perspectives, 31(2), 211-36.
3. https://www.kaggle.com/code/shivamburnwal/nlp-98-acc-eda-with-model-using-spacy-pipeline
4. https://www.kaggle.com/code/madz2000/text-classification-using-keras-nb-97-accuracy

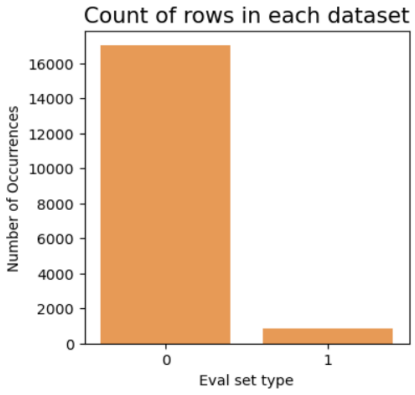
Milestone-2

Index:

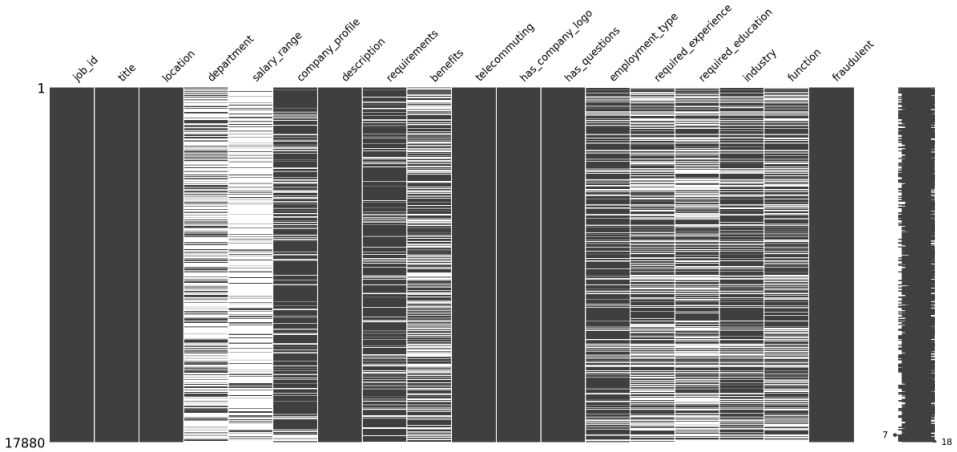
1. Prior Work Evaluation:

Based on our work in previous milestone, we had re-evaluated the work and identified that the basic pre-processing of data such as identifying the irrelevant columnar data, columns with more null values, filling blank values with null/NaN, filtering the data had not been performed which might have resulted in misleading the result and accuracy of our data in current and future works. As a learning factor, we focused more on pre-processing the data to fix the blank, null or NaN values and then also performed Exploratory Data Analysis (EDA). During our second phase of work, we realized the value and scope of work that we have with the Real/Fake Job posting dataset which helped to expand our ideology to work on both kinds of Feature Selection Methods such as Numerical Variables and also Categorical Variables. With the help of EDA, we were able to improve our pre-processing/cleaning the actual dataset and then proceeded with implementation of different Feature Selection Techniques.

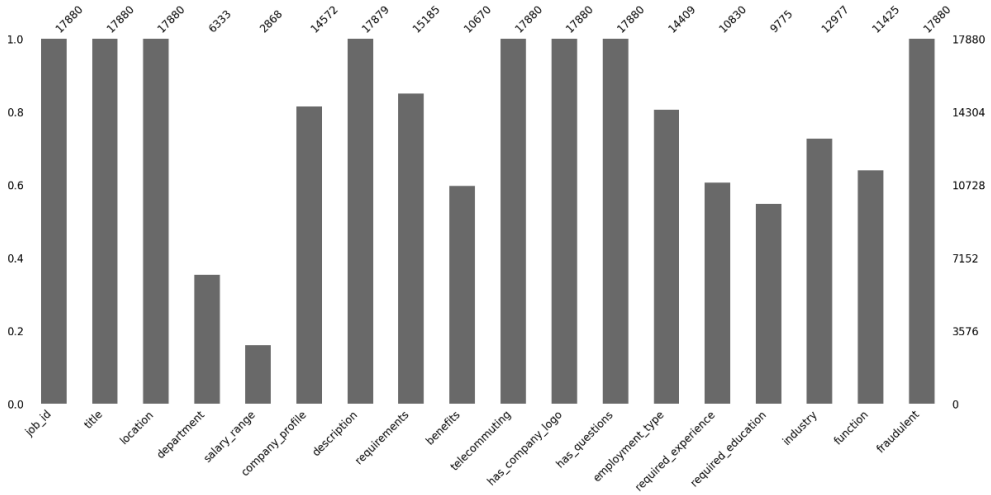
- Screenshot below shows the total number of real job postings on the left and fake postings on the right bar.



- Screenshot below is the visualization of missing values in each column in the dataset.



- Screenshot below is the visualization of bar plot of each column against missing values.



2. Exploratory Data Analysis:

Exploratory Data Analysis (EDA) is a crucial step in the data analysis process that involves examining and understanding the dataset's characteristics, structure, and relationships between variables. Listed down are some of the key aspects that we followed during the EDA phase of processing the data.

i. Loaded and previewed the data for the datatype of each column to verify if there is any mismatch.

ii. By following this method, we were able to identify numerical and categorical variables present in the dataset.

iii. We learnt the different types of visualizations that can be plotted for both numerical and categorical features which helps to identify and summarize the statistics and frequency tables, and also the relationships between variables. By doing so, we were able to understand the dataset better.

- Screenshots below are sample code showing exploratory data analysis performed for both numerical and categorial variables.

Exploratory Data analysis for Numerical Variables

In [14]:
```
# preapre a list of cloumns which can be deleted

length_of_dataset = len(data)
drop_list = list()
for key,value in null_columns.items():
    if value > int(length_of_dataset * 10 /100):
        print(key,":",value)
        drop_list.append(key)
```

```
department : 11547
salary_range : 15012
company_profile : 3308
requirements : 2695
benefits : 7210
employment_type : 3471
required_experience : 7050
required_education : 8105
industry : 4903
function : 6455
```

In [15]:
```
#Dropping null valued columns

for values in drop_list:
    data = data.drop(values,axis = 1)
```

```
null_list = list(null_columns.keys())
remaining_list = [item for item in null_list if item not in drop_list]
print(remaining_list)
```

```
['description']
```

In [18]:
```
data = data.dropna()
```

In [19]:
```
data.isna().sum()
```

Out[19]:
```
job_id              0
title               0
location            0
description         0
telecommuting       0
has_company_logo    0
has_questions       0
fraudulent          0
dtype: int64
```

In [20]:
```
mapping = {k: v for v, k in enumerate(data.title.unique())}
data['title'] = data.title.map(mapping)
data.head()
```

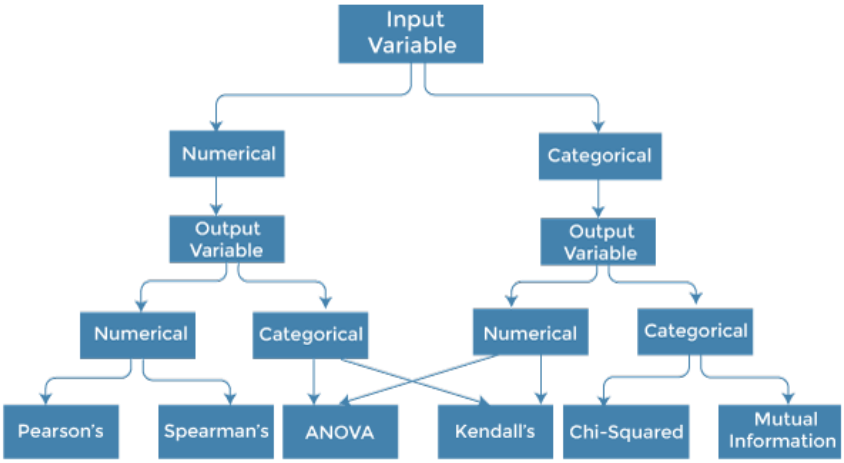
Out[20]:

	job_id	title	location	description	telecommuting	has_company_logo	has_questions	fraudulent
0	1	0	US, NY, New York	Food52, a fast-growing, James Beard Award-winn...	0	1	0	0
1	2	1	NZ, , Auckland	Organised - Focused - Vibrant - Awesome!Do you...	0	1	0	0
2	3	2	US, IA, Wever	Our client, located in Houston, is actively se...	0	1	0	0
3	4	3	US, DC, Washington	THE COMPANY: ESRI – Environmental Systems Rese...	0	1	0	0
4	5	4	US, FL, Fort Worth	JOB TITLE: Itemization Review ManagerLOCATION:...	0	1	1	0

In [21]:
```
mapping = {k: v for v, k in enumerate(data.location.unique())}
data['location'] = data.location.map(mapping)
data.head()
```

3. Feature Selection Techniques:

Feature selection is the process of identifying the most important features (i.e., independent variables or predictors) in a dataset which can explain the variation in the dependent variable or target variable. In datasets with numerical variables, there are various techniques for feature selection. we can use these techniques to improve the accuracy and interpretability of the models.

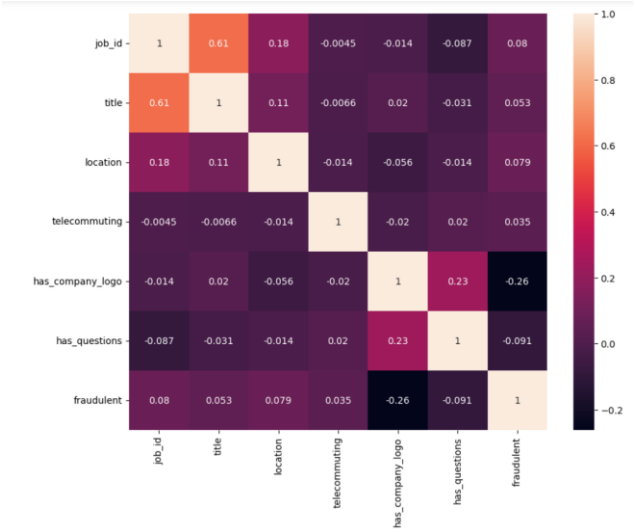


Classification of Feature Selection based on variables

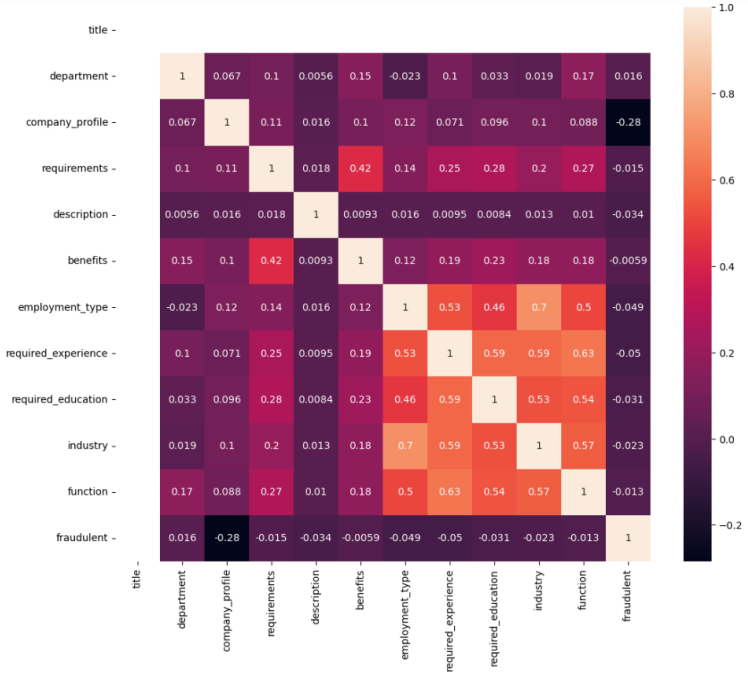
i. Correlation analysis:

Correlation analysis is a technique that helps to identify the relationship between two numerical variables. We used this technique to identify the variables that have a strong correlation with the target variable. we calculated the correlation coefficient between each numerical variable and the target variable and selected the variables with the highest correlation coefficients. Correlation analysis has been performed and the heat maps for numerical and categorical variables were plotted which provided greater visibility of target variables.

- Screenshot below shows the correlation heat map for numerical variables after EDA.



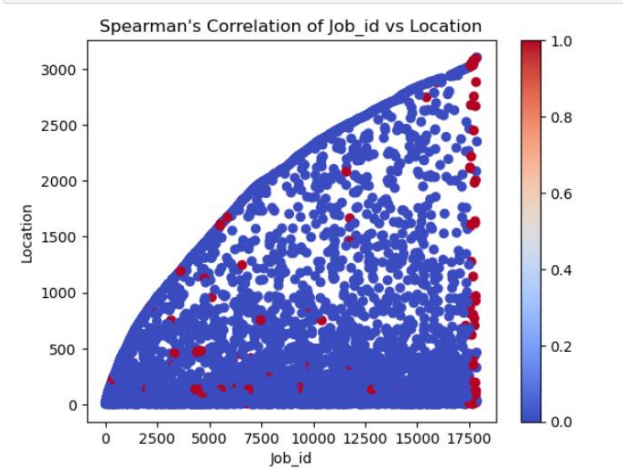
- Screenshot below shows the correlation heat map for categorical variables after EDA.



ii. Spearman's Correlation:

Spearman's correlation is based on ranks rather than the actual values of the variables. It assesses the strength and direction of the relationship between two numerical variables. The ranks of each variable were calculated, and then the correlation coefficient is calculated based on these ranks. The coefficient ranges from -1 to +1, where -1 indicates a perfectly negative correlation, +1 indicates a perfectly positive correlation, and 0 indicates no correlation between the variables. Spearman's correlation can be useful when we have data that is not normally distributed or when the relationship between the variables is non-linear.

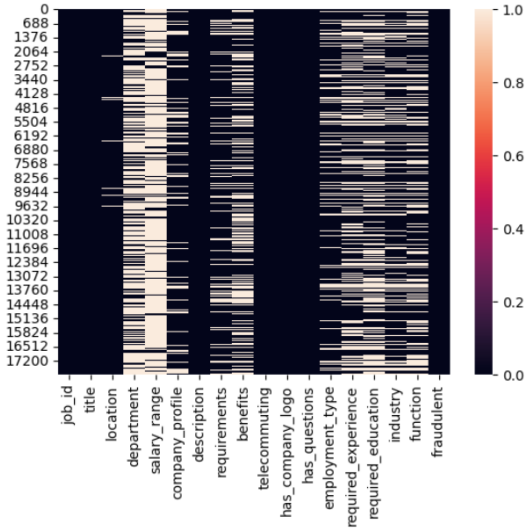
- Screenshot below shows the Spearman's Correlation for 2 variables **job_id vs location**



iii. Text Cleaning to build world cloud:

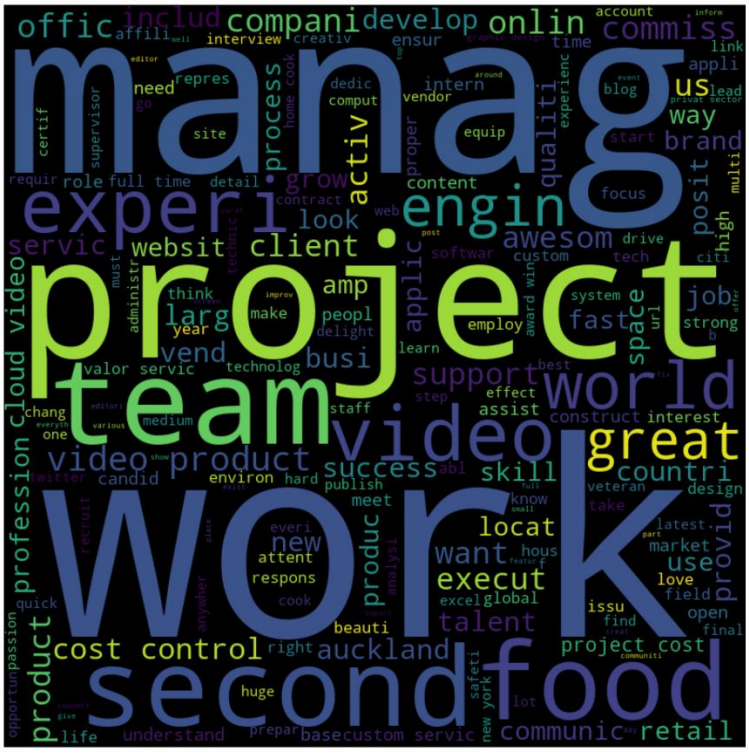
Text cleaning is the process of preparing text data for analysis by removing irrelevant or unwanted information, such as stop words, punctuation, and special characters, and converting the text to a standard format. We performed this step to clean the categorical variables by removing HTML tags, punctuation, and special characters, stop words etc. This helped in building a word cloud, and we plotted the most frequently occurring words in a text corpus with both real and fake job postings. The size of each word in the cloud corresponds to its frequency in the corpus.

- Screenshot below shows the overall missing values present in categorical variables in dataset.



- Screenshot below shows the world cloud for most used words detected in real job posting.

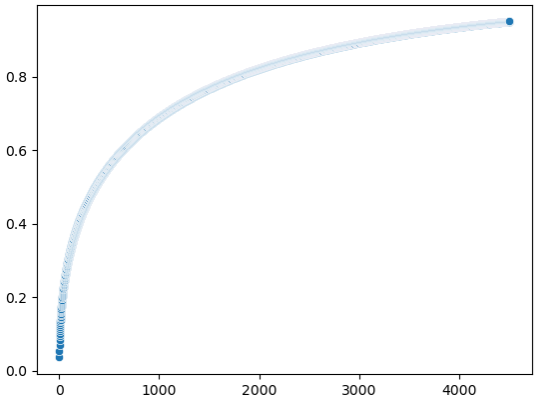
- Screenshot below shows the world cloud for most used words detected in fake job posting.



iv. Principal Component Analysis (PCA):

PCA is a dimensionality reduction technique which we used to identify the most important features in a dataset. It transformed the original numerical variables into a new set of variables that capture the most important features. The new variables, called principal components, are linear combinations of the original variables. We performed PCA to reduce the categorical dimensions of the data set and the plots were shown in the code below.

- Screenshot below shows the PCA curve plotted for reduction of dimensions in the dataset for categorical target variable “fraudulent”



4. Future work:

With our current work, we were able to understand different techniques in pre-processing and feature selection which helped us to identify the target variables. The next step is to split the data into training and testing sets. The training set is used to train the model, while the testing set is used to evaluate its performance. Post which we can use the selected features to build a machine learning model. The model can be used for prediction, classification, or clustering, depending on the problem at hand. This can vary depending on the nature of the data and the task at hand. Some common machine learning algorithms include linear regression, logistic regression, decision trees, random forests, support vector machines, and neural networks.

Once the model has been built and trained on the training data, we can evaluate its performance on the testing data. This can be done by computing various metrics, such as accuracy, precision, recall, and F1 score, depending on the problem.

5. References:

- Ahmed, H., Traore, I., & Saad, S. (2017). Detection of online fake news using n-gram analysis and machine learning techniques. In International conference on intelligent, secure, and dependable systems in distributed and cloud environments, 127-138.
- Allcott, H., & Gentzkow, M. (2017). Social media and fake news in the 2016 election. Journal of economic perspectives, 31(2), 211-36.
- https://www.kaggle.com/code/shivamburnwal/nlp-98-acc-eda-with-model-using-spacy-pipeline
- https://www.kaggle.com/code/madz2000/text-classification-using-keras-nb-97-accuracy
- How to Choose a Feature Selection Method For Machine Learning:
- Feature Selection Techniques in Machine Learning
- Fake Job Posting Detection and Getting Useful Insights from the Job Postings
- https://github.com/sharad18/Fake_Job_Posting/blob/master/Data%20Cleaning%20%26%20EDA.ipynb