CHAPTER-1

COMPANY PROFILE

1.1 INTRODUCTION

YHills is an innovative platform committed to revolutionizing education and training for college students and fresh graduates. At YHills, we understand the challenges faced by individuals entering the workforce, particularly the gap between academic learning and industry requirements. With this in mind, they have developed a comprehensive approach to education that combines theoretical knowledge with practical skills, ensuring that learners are well-equipped to succeed in their careers.

Figure 1.1 Company Logo

1.2 OVERVIEW

YHills is revolutionizing the educational landscape by addressing the prevailing mismatch between graduates' skills and industry demands. With features such as one-year course access, live classes, mentorship, capstone projects, internship certificates, and course completion certificates, we ensure our students stay ahead of the curve and achieve their career goals.

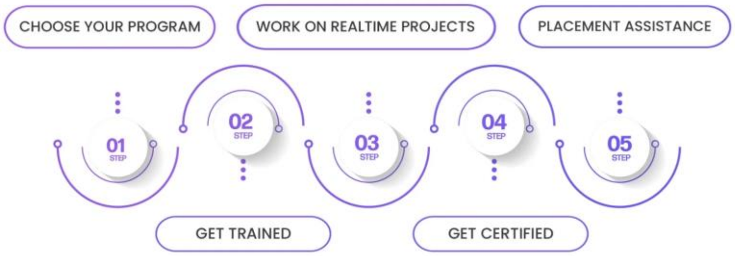


Figure 1.2 Overview of the Company

1.3 MISSION AND VISION OF THE COMPANY

YHills aims to redefine the employment scenario by empowering learners to make meaningful contributions to society. Our mission is to bridge the gap between traditional education and the demands of the digital age, fostering individual and collaborative growth. Rooted in the philosophy of "Teaching a Man How to Fish," our vision is to enable learners to hone their skills, contribute meaningfully, and excel in their chosen fields.

1.4 TEAM

At YHills, our success is driven by our diverse and talented team. Led by visionary founders Aman Kumar and Mani Bhushan, our team comprises passionate individuals committed to transforming education through technology. From our CEO and Chief Business Officer to our dedicated team leaders and operation specialists, each member brings unique expertise and a shared passion for innovation.

Founder & CEO

Aman Kumar

Founder & Chief Business Officer

Mani Bhushan

Figure 1.3 Founder and Chief Business Officer of YHills

1.5 SERVICES AND CLIENTS

YHills takes pride in offering a range of services tailored to meet the diverse needs of our students. With a focus on live classes, mentorship, and hands-on projects, we ensure our learners receive comprehensive training that prepares them for real-world challenges. Our partnerships with leading organizations and our alumni's success stories at esteemed companies such as Oracle, Amazon, and Walmart underscore our commitment to excellence and the tangible impact of our programs on students' career trajectories.

CHAPTER-2

ABOUT THE COMPANY

YHills is a pioneering educational platform committed to revolutionizing the way college students and fresh graduates acquire essential skills for the modern workforce. Founded by Aman Kumar and Mani Bhushan, YHills was born out of a shared vision to address the glaring gap between traditional education and industry requirements.

At YHills, they believe that education is the cornerstone of personal and professional growth. However, they recognized the need for a more practical and relevant approach to learning, one that goes beyond textbooks and lectures to provide students with the skills and knowledge they need to succeed in today's dynamic job market.

Driven by their vision, they have developed a comprehensive platform that offers customized and cost-effective training programs tailored to meet the needs of learners at every stage of their academic and professional journey. Their courses cover a wide range of disciplines, from technology and business to design and marketing, ensuring that their learners are well-equipped to pursue their passions and achieve their goals.

What sets YHills apart is their commitment to excellence and innovation in education. They leverage cutting-edge technology and teaching methodologies to deliver engaging and interactive learning experiences that inspire creativity, critical thinking, and problem-solving skills. Their team of expert instructors, mentors, and industry professionals is dedicated to providing personalized guidance and support to help learners reach their full potential.

At YHills, they are not just in the business of education; they are in the business of transforming lives. They believe in the power of education to unlock human potential and drive positive change in the world. By empowering individuals with the skills and knowledge they need to succeed, they are building a brighter future for generations to come.

CHAPTER -3

TASKS PERFORMED

Data Scientists are responsible for leveraging data to derive insights and make informed decisions. They work with cross-functional teams to develop data-driven solutions that address business challenges and improve decision-making processes.

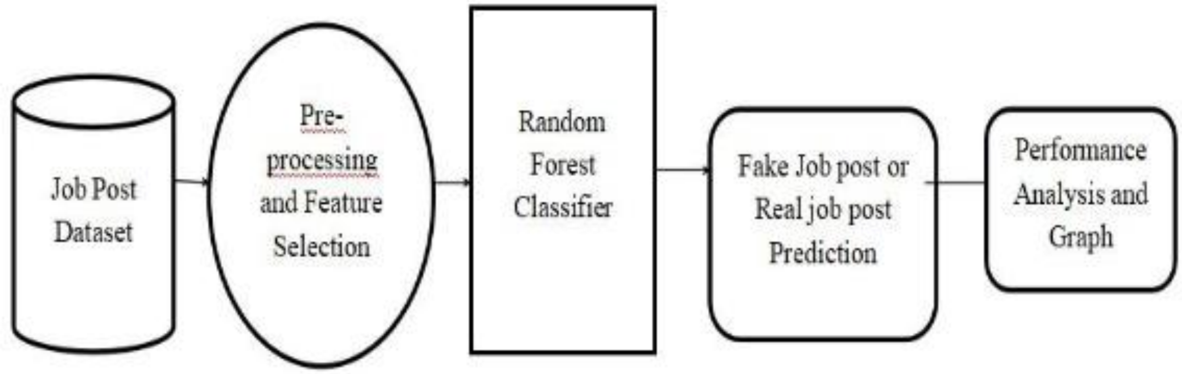

Job Post Dataset
Pre-processing and Feature Selection
Random Forest Classifier
Fake Job post or Real job post Prediction
Performance Analysis and Graph

Figure 3.1 System Architecture

The above figure 3.1 represent system architecture of fraudulent job prediction. The job dataset provided undergoes preprocessing and feature selection before being used to train a Random Forest classifier, which predicts whether a job posting is fake or real. Following the prediction, performance analysis is conducted, and graphs are generated to illustrate the classifier's accuracy and other relevant metrics. The responsibilities for this tasks includes,

- Cleaning and preparing the job dataset.
- Identifying relevant features for the Random Forest classifier.
- Training the classifier to predict job authenticity.
- Evaluating classifier performance using relevant metrics.
- Creating visualizations to represent classifier performance.
- Summarizing findings in a clear and concise manner.

3.1 SERVICES/TOOLS/PROGRAMMING LANGUAGES USED

The project utilized a variety of services, tools, languages, and models for detecting fraudulent job postings and analyzing datasets. These resources were crucial for statistical analysis, visualization, and model development.

Services:

➢ **Jupyter Notebook**

- **Primary Environment:** Jupyter Notebook served as the primary environment for implementing the fraudulent job prediction project. It facilitated interactive data analysis and code execution, allowing seamless integration of code, visualizations, and narrative text within a single document.
- **Interactivity:** Jupyter Notebook's interactive nature enabled iterative development and experimentation, facilitating rapid prototyping and model refinement.
- **Visualization:** Through Jupyter Notebook, visualizations such as charts, graphs, and statistical summaries were created to explore data patterns and present findings effectively.

Tools:

➢ **Kaggle**

- **Data Collection:** Kaggle was instrumental in collecting datasets related to job postings and fraudulent activities. It provided access to a diverse range of publicly available datasets and competitions, offering a rich source of data for analysis.
- **Dataset Repository:** Kaggle's extensive dataset repository offered datasets specifically curated for data science projects, including those related to job postings, which were crucial for training and evaluating predictive models.

Programming Languages:

➢ **Python**

- **Data Analysis:** Python served as the primary programming language for data analysis, model development, and visualization within the Jupyter Notebook environment.

- **Versatility:** Python's versatility and rich ecosystem of libraries made it well-suited for various data science tasks, including data manipulation, statistical analysis, and machine learning.

➢ **Libraries:**

1. **Pandas:** Used for data manipulation and preprocessing tasks, such as cleaning and transforming datasets obtained from Kaggle.
2. **NumPy:** Utilized for numerical computing tasks, providing support for mathematical operations and array manipulation.
3. **Scikit-learn:** Leveraged for implementing machine learning models, including the Random Forest classifier used for detecting fraudulent job postings.
4. **Matplotlib:** Employed for creating visualizations, such as charts and statistics, to analyze data patterns and present findings.
5. **Other Python Libraries:** Depending on specific analysis requirements, additional libraries may have been imported for tasks such as feature engineering, model evaluation, and hyper parameter tuning.

Model:

➢ **Random Forest Classifier**

- Chosen for handling complex datasets, Random Forest builds multiple decision trees and combines their predictions to reduce overfitting.
- Its robustness is evident in its ability to be less sensitive to noise and outliers compared to other algorithms.
- Hyper-parameters, set prior to training, control the algorithm's behavior and remain constant throughout.
- Examples include the number of trees, maximum tree depth, and minimum samples required to split a node in Random Forest.

3.2 Implementation

The implementation phase marks the transition from theoretical design to a programmable format. Here, the application is divided into distinct modules and coded for deployment. Jupyter Notebook serves as the front end, while the fake job prediction dataset acts as the back-end database. Python is chosen as the programming language for this implementation.

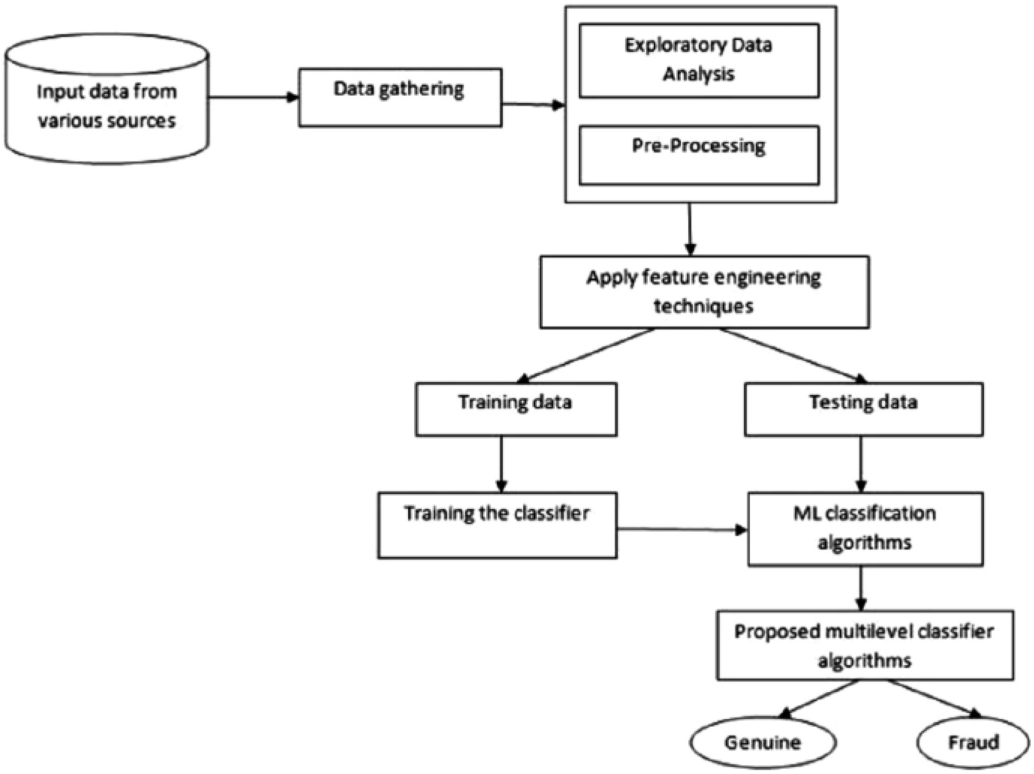

Input data from various sources
Data gathering
Exploratory Data Analysis
Pre-Processing
Apply feature engineering techniques
Training data
Testing data
Training the classifier
ML classification algorithms
Proposed multilevel classifier algorithms
Genuine
Fraud

Figure 3.2 Implementation of Model

The above figure 3.2 shows implementation of machine learning model to predict fake jobs where, input data is collected from diverse sources and undergoes exploratory data analysis and preprocessing stages. Following successful completion of these steps, feature engineering techniques, such as creating new features or transforming existing ones, are applied to enhance the data. For instance, in the context of job fraudulent prediction, features like the frequency of certain keywords in job descriptions or the length of job titles could be engineered to improve the model's predictive performance. Subsequently, the dataset is split into training and test sets. The training data is utilized to train the machine learning classification algorithm, while the test data is directly fed into the algorithm for prediction. Finally, a proposed multilevel classifier algorithm, utilizing the Random Forest model, is employed to classify job postings as either fake or genuine. The application is structured into primary modules:

1. **Load Dataset Module:**

   This module focuses on loading the dataset collected from Kaggle and providing its information as input to subsequent modules.

2. **Generate Test and Train Module:**

   This module divides the dataset into test and train datasets using a 70:30 ratio. Seventy percent of the data is utilized for training the system, while thirty percent is reserved for testing the model performance.

3. **Random Forest Algorithm Execution:**

   Multiple decision trees are trained independently on random subsets of the training data. Each decision tree is trained using a technique called bootstrap aggregation or "bagging." During training, each tree predicts the target variable based on a random subset of features, ensuring diversity among trees. When testing, the predictions of all individual trees are combined through a process called "voting" or "averaging."

   For classification tasks, the mode (most frequent prediction) of individual tree predictions is taken as the final prediction. For regression tasks, the average of individual tree predictions is taken as the final prediction. This approach ensures robust performance and generalization of the model, allowing for effective prediction of fraudulent job postings.

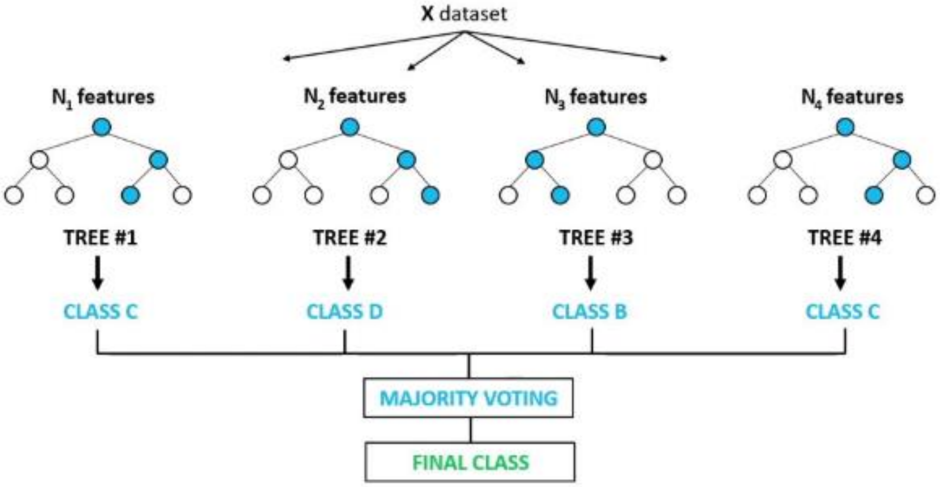
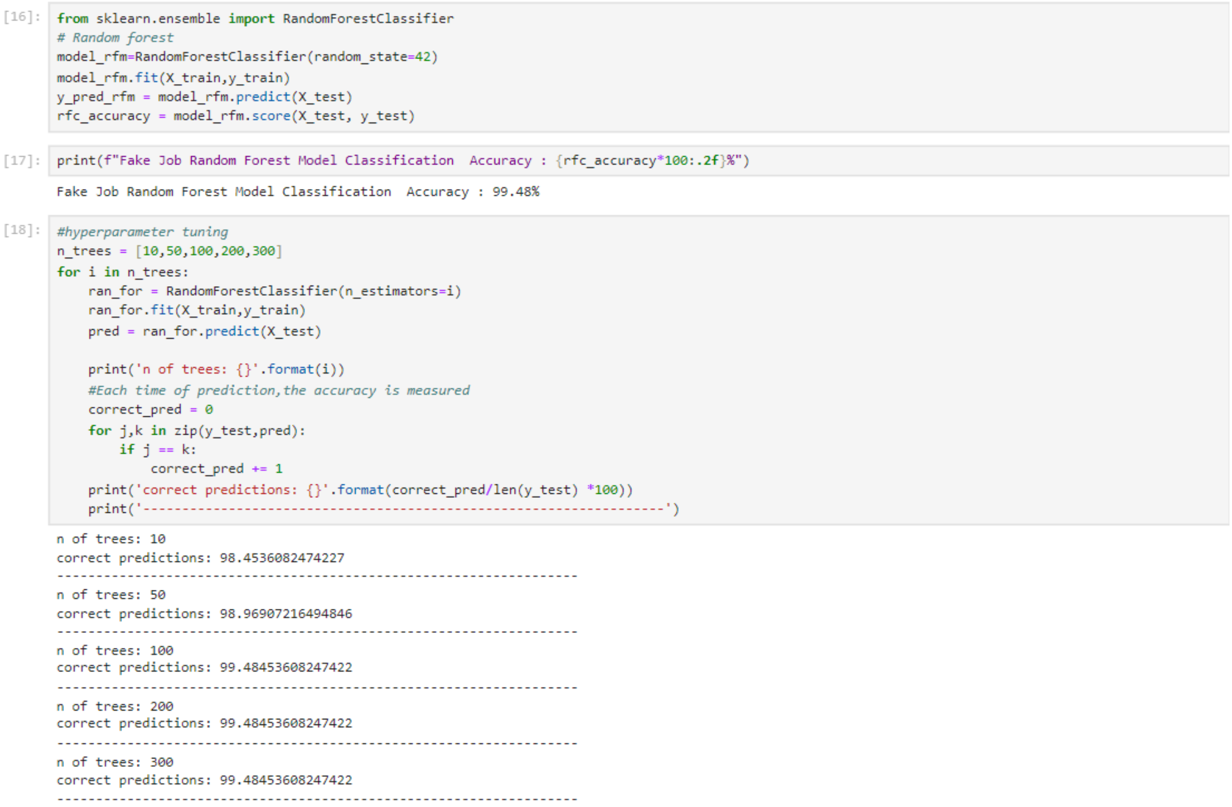


Figure 3.3 Simplified Random Forest Model

4. **Fraudulent Job Prediction:** Multiple algorithms are applied to detect fraudulent job postings within the test data. The model categorizes the data into genuine and fraudulent job postings. The Random Forest algorithm exhibits superior accuracy in identifying fraudulent activities compared to other algorithms like Logistic Regression, SVM, and Naïve Bayes.



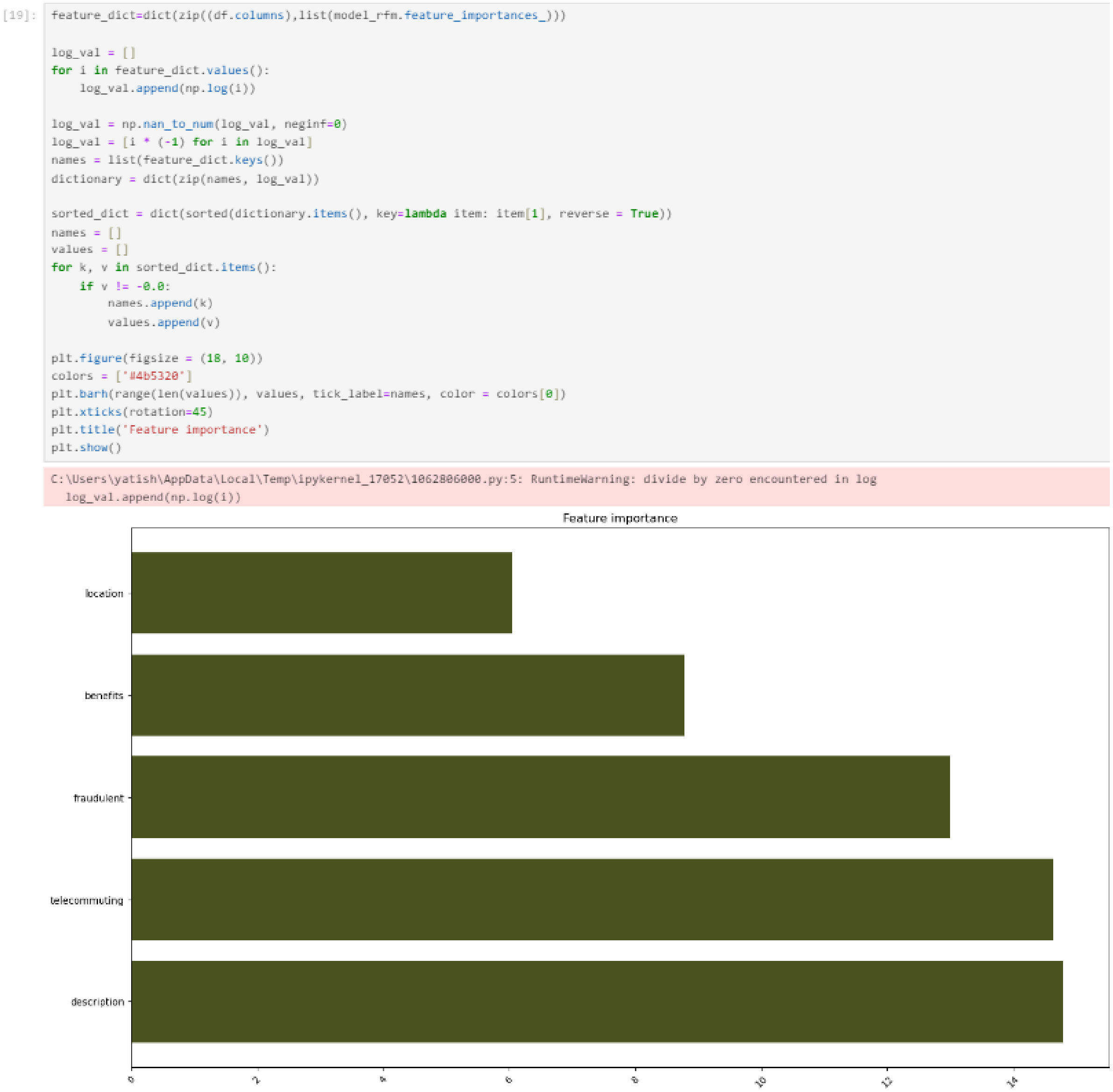
```
[16]: from sklearn.ensemble import RandomForestClassifier
      # Random forest
      model_rfm=RandomForestClassifier(random_state=42)
      model_rfm.fit(X_train,y_train)
      y_pred_rfm = model_rfm.predict(X_test)
      rfc_accuracy = model_rfm.score(X_test, y_test)

[17]: print(f"Fake Job Random Forest Model Classification  Accuracy : {rfc_accuracy*100:.2f}%")

      Fake Job Random Forest Model Classification  Accuracy : 99.48%

[18]: #hyperparameter tuning
      n_trees = [10,50,100,200,300]
      for i in n_trees:
          ran_for = RandomForestClassifier(n_estimators=i)
          ran_for.fit(X_train,y_train)
          pred = ran_for.predict(X_test)

          print('n of trees: {}'.format(i))
          #Each time of prediction,the accuracy is measured
          correct_pred = 0
          for j,k in zip(y_test,pred):
              if j == k:
                  correct_pred += 1
          print('correct predictions: {}'.format(correct_pred/len(y_test) *100))
          print('-------------------------------------------------------------------')

      n of trees: 10
      correct predictions: 98.4536082474227
      -------------------------------------------------------------------
      n of trees: 50
      correct predictions: 98.96907216494846
      -------------------------------------------------------------------
      n of trees: 100
      correct predictions: 99.48453608247422
      -------------------------------------------------------------------
      n of trees: 200
      correct predictions: 99.48453608247422
      -------------------------------------------------------------------
      n of trees: 300
      correct predictions: 99.48453608247422
      -------------------------------------------------------------------
```

Figure 3.4 Implementation of Random Forest Classifier

The Figure 3.3 represents the importing of Random Forest Classifier from SKLearn module which shows accuracy percentage of the module and hyper parameter tuning to improve its performance.

5. **Comparative Analysis:**

The dataset is rigorously tested with various algorithms such as Logistic Regression, Random Forest, SVM, and Naïve Bayes. Among these, Random Forest emerges as the most adept in achieving accuracy, surpassing its counterparts. The implementation of libraries such as NumPy, Pandas, and Matplotlib further enhances the efficiency and effectiveness of the project.



```
[19]: feature_dict=dict(zip((df.columns),list(model_rfm.feature_importances_)))

      log_val = []
      for i in feature_dict.values():
          log_val.append(np.log(i))

      log_val = np.nan_to_num(log_val, neginf=0)
      log_val = [i * (-1) for i in log_val]
      names = list(feature_dict.keys())
      dictionary = dict(zip(names, log_val))

      sorted_dict = dict(sorted(dictionary.items(), key=lambda item: item[1], reverse = True))
      names = []
      values = []
      for k, v in sorted_dict.items():
          if v != -0.0:
              names.append(k)
              values.append(v)

      plt.figure(figsize = (18, 10))
      colors = ['#4b5320']
      plt.barh(range(len(values)), values, tick_label=names, color = colors[0])
      plt.xticks(rotation=45)
      plt.title('Feature importance')
      plt.show()
```

```
C:\Users\yatish\AppData\Local\Temp\ipykernel_17052\1062806000.py:5: RuntimeWarning: divide by zero encountered in log
  log_val.append(np.log(i))
```

Figure 3.5 Feature Importance

The figure 3.4 represents feature importance to determine most influential feature in predictive model

CHAPTER -4

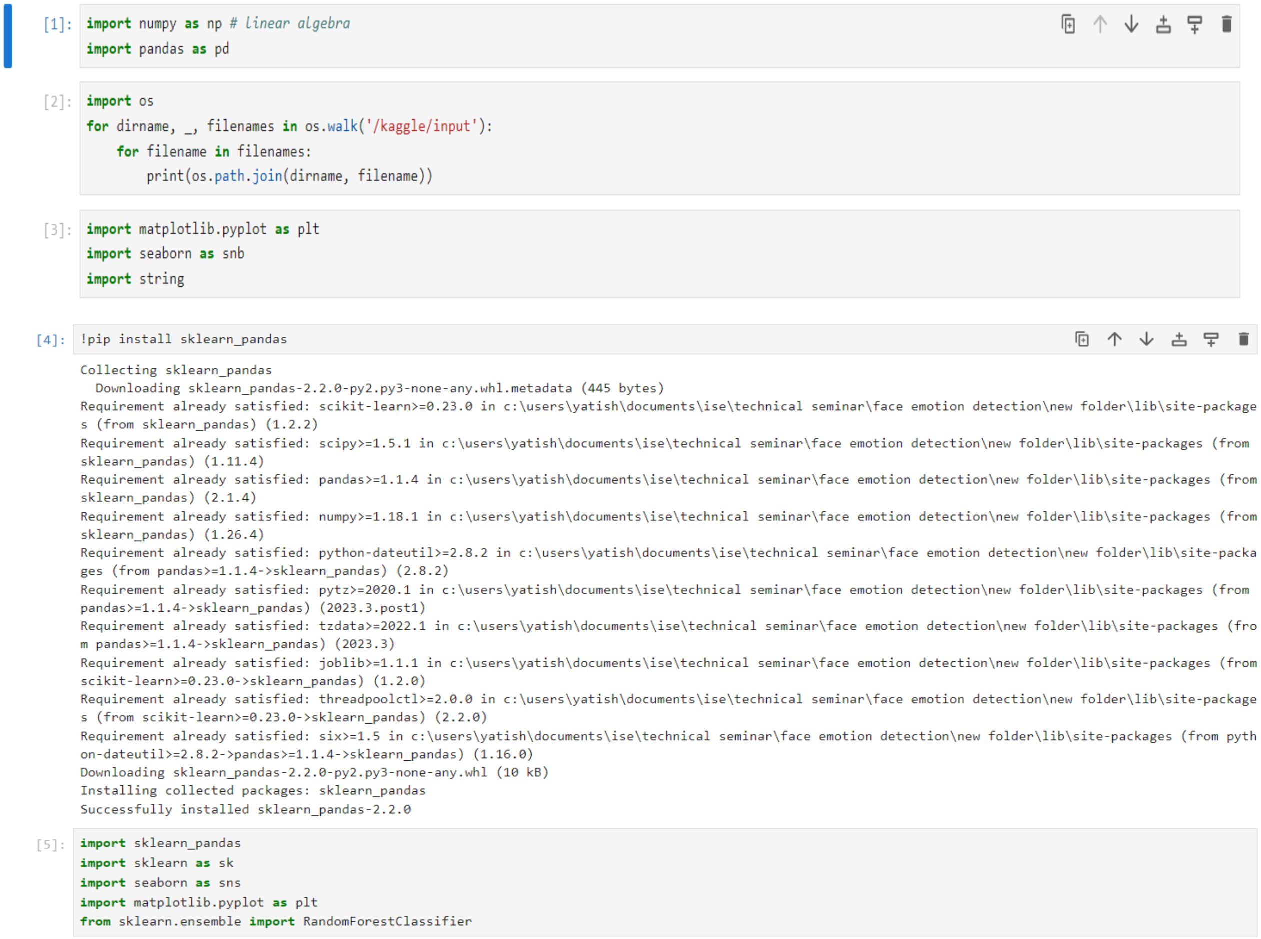
SPECIFIC OUTCOMES

In this section we try to design our current model using Python as programming language and we used Jupyter NoteBook as working environment for executing the application.

4.1 TECHNICAL

1. IMPORT LIBRARIES

Importing essential Python libraries such as Pandas, NumPy, Matplotlib, and Scikit-learn to facilitate data manipulation, analysis, visualization, and machine learning model implementation within the Jupyter Notebook environment.



```
[1]: import numpy as np # linear algebra
     import pandas as pd

[2]: import os
     for dirname, _, filenames in os.walk('/kaggle/input'):
         for filename in filenames:
             print(os.path.join(dirname, filename))

[3]: import matplotlib.pyplot as plt
     import seaborn as snb
     import string

[4]: !pip install sklearn_pandas
```

```
Collecting sklearn_pandas
  Downloading sklearn_pandas-2.2.0-py2.py3-none-any.whl.metadata (445 bytes)
Requirement already satisfied: scikit-learn>=0.23.0 in c:\users\yatish\documents\ise\technical seminar\face emotion detection\new folder\lib\site-package
s (from sklearn_pandas) (1.2.2)
Requirement already satisfied: scipy>=1.5.1 in c:\users\yatish\documents\ise\technical seminar\face emotion detection\new folder\lib\site-packages (from
sklearn_pandas) (1.11.4)
Requirement already satisfied: pandas>=1.1.4 in c:\users\yatish\documents\ise\technical seminar\face emotion detection\new folder\lib\site-packages (from
sklearn_pandas) (2.1.4)
Requirement already satisfied: numpy>=1.18.1 in c:\users\yatish\documents\ise\technical seminar\face emotion detection\new folder\lib\site-packages (from
sklearn_pandas) (1.26.4)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\yatish\documents\ise\technical seminar\face emotion detection\new folder\lib\site-packa
ges (from pandas>=1.1.4->sklearn_pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in c:\users\yatish\documents\ise\technical seminar\face emotion detection\new folder\lib\site-packages (from
pandas>=1.1.4->sklearn_pandas) (2023.3.post1)
Requirement already satisfied: tzdata>=2022.1 in c:\users\yatish\documents\ise\technical seminar\face emotion detection\new folder\lib\site-packages (fro
m pandas>=1.1.4->sklearn_pandas) (2023.3)
Requirement already satisfied: joblib>=1.1.1 in c:\users\yatish\documents\ise\technical seminar\face emotion detection\new folder\lib\site-packages (from
scikit-learn>=0.23.0->sklearn_pandas) (1.2.0)
Requirement already satisfied: threadpoolctl>=2.0.0 in c:\users\yatish\documents\ise\technical seminar\face emotion detection\new folder\lib\site-package
s (from scikit-learn>=0.23.0->sklearn_pandas) (2.2.0)
Requirement already satisfied: six>=1.5 in c:\users\yatish\documents\ise\technical seminar\face emotion detection\new folder\lib\site-packages (from pyth
on-dateutil>=2.8.2->pandas>=1.1.4->sklearn_pandas) (1.16.0)
Downloading sklearn_pandas-2.2.0-py2.py3-none-any.whl (10 kB)
Installing collected packages: sklearn_pandas
Successfully installed sklearn_pandas-2.2.0
```

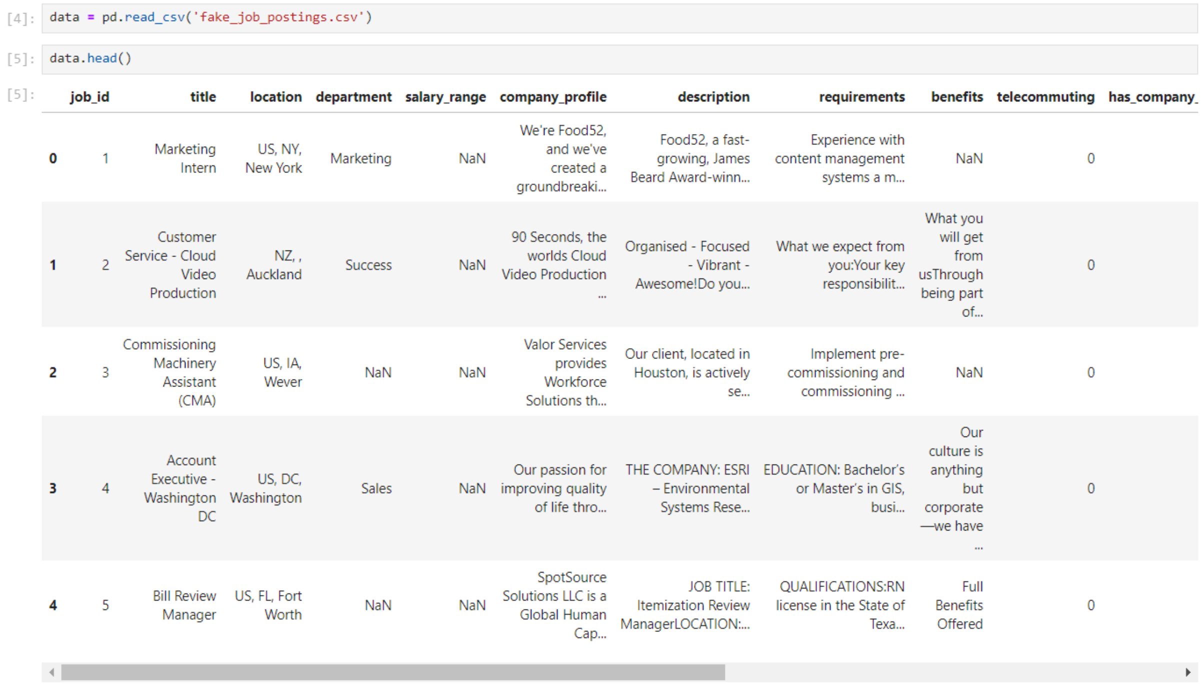
```
[5]: import sklearn_pandas
     import sklearn as sk
     import seaborn as sns
     import matplotlib.pyplot as plt
     from sklearn.ensemble import RandomForestClassifier
```

Figure 4.1 Imported libraries and modules

The figure 4.1 above window clearly represents the list of several modules used in our project.

2. PRE-PROCESS THE DATA

Implementing data cleaning and transformation techniques to prepare the dataset for analysis, including handling missing values, removing duplicates, and standardizing data formats.



```
[4]: data = pd.read_csv('fake_job_postings.csv')
```

```
[5]: data.head()
```

[5]:

	job_id	title	location	department	salary_range	company_profile	description	requirements	benefits	telecommuting	has_company_
0	1	Marketing Intern	US, NY, New York	Marketing	NaN	We're Food52, and we've created a groundbreaki...	Food52, a fast-growing, James Beard Award-winn...	Experience with content management systems a m...	NaN	0	
1	2	Customer Service - Cloud Video Production	NZ, , Auckland	Success	NaN	90 Seconds, the worlds Cloud Video Production ...	Organised - Focused - Vibrant - Awesome!Do you...	What we expect from you:Your key responsibilit...	What you will get from usThrough being part of...	0	
2	3	Commissioning Machinery Assistant (CMA)	US, IA, Wever	NaN	NaN	Valor Services provides Workforce Solutions th...	Our client, located in Houston, is actively se...	Implement pre-commissioning and commissioning ...	NaN	0	
3	4	Account Executive - Washington DC	US, DC, Washington	Sales	NaN	Our passion for improving quality of life thro...	THE COMPANY: ESRI – Environmental Systems Rese...	EDUCATION: Bachelor's or Master's in GIS, busi...	Our culture is anything but corporate—we have ...	0	
4	5	Bill Review Manager	US, FL, Fort Worth	NaN	NaN	SpotSource Solutions LLC is a Global Human Cap...	JOB TITLE: Itemization Review ManagerLOCATION:...	QUALIFICATIONS:RN license in the State of Texa...	Full Benefits Offered	0	

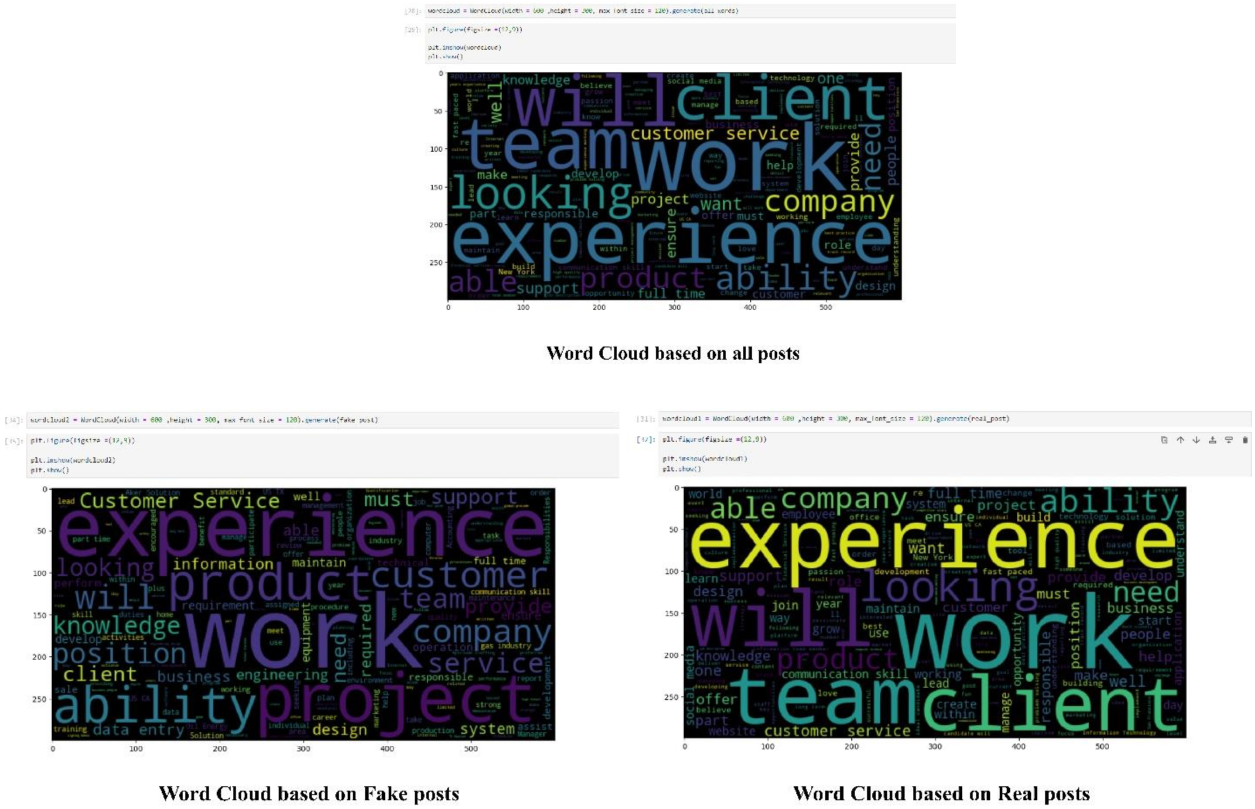
Figure 4.2 Pre- processed datasets

The figure 4.2 above window we can see data is pre-processed and we converted dataset into test and train.

3. WORD CLOUD

Text data visualization techniques, such as word clouds, offer valuable insights into the most frequent terms and phrases within a dataset of job postings. By analyzing the textual content, we can identify common keywords and patterns associated with fraudulent activity, aiding in the detection and prevention of fraudulent job postings.

Word clouds visually represent the frequency of words in a dataset by varying the size and color of each word based on its frequency of occurrence. Words that appear more frequently are displayed in larger font sizes, while less common words are smaller. Additionally, colors can be used to further emphasize word frequency or to categorize words based on certain criteria.



```
[28]: wordcloud = WordCloud(width = 600 ,height = 300, max_font_size = 120).generate(all_words)

[29]: plt.figure(figsize =(12,9))

      plt.imshow(wordcloud)
      plt.show()
```

Word Cloud based on all posts

```
[34]: wordcloud2 = WordCloud(width = 600 ,height = 300, max_font_size = 120).generate(fake_post)

[35]: plt.figure(figsize =(12,9))

      plt.imshow(wordcloud2)
      plt.show()
```

Word Cloud based on Fake posts

```
[31]: wordcloud1 = WordCloud(width = 600 ,height = 300, max_font_size = 120).generate(real_post)

[32]: plt.figure(figsize =(12,9))

      plt.imshow(wordcloud1)
      plt.show()
```

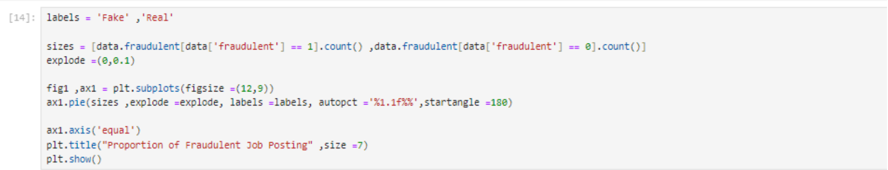
Word Cloud based on Real posts

Figure 4.3 Word cloud on all, real, fake posts

The above figure 4.3, 4.4, 4.5 represents word cloud based on all, real, fake posts. By visualizing text data in this way, we can gain a better understanding of the language used in fraudulent job postings, enabling us to develop more effective strategies for detection and mitigation. Additionally, word clouds serve as a powerful tool for communicating findings to stakeholders and decision-makers in a clear and engaging manner.

4. PERFORMANCE ANALYSIS

Performing an in-depth evaluation of the model's performance metrics to gauge its accuracy and efficacy in distinguishing between fraudulent and legitimate job postings across various dimensions, including categories such as fake and real job posts, geographical regions, job experience levels, presence of company logos, and the prevalence of fraudulent activities.



```
[14]: labels = 'Fake' ,'Real'

      sizes = [data.fraudulent[data['fraudulent'] == 1].count() ,data.fraudulent[data['fraudulent'] == 0].count()]
      explode =(0,0.1)

      fig1 ,ax1 = plt.subplots(figsize =(12,9))
      ax1.pie(sizes ,explode =explode, labels =labels, autopct ='%1.1f%%',startangle =180)

      ax1.axis('equal')
      plt.title("Proportion of Fraudulent Job Posting" ,size =7)
      plt.show()
```

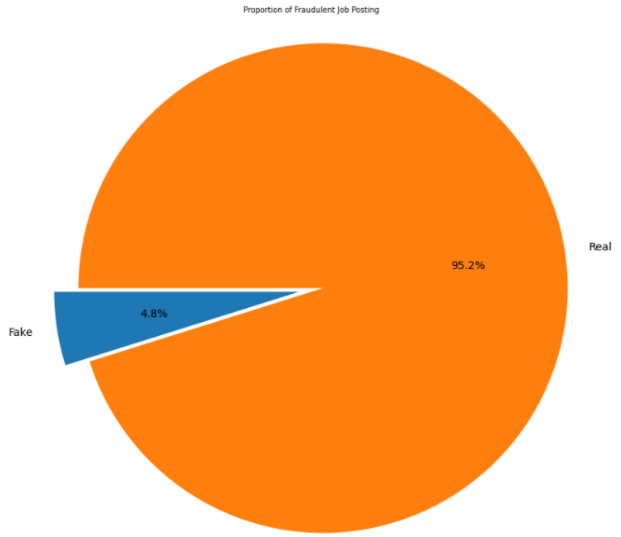
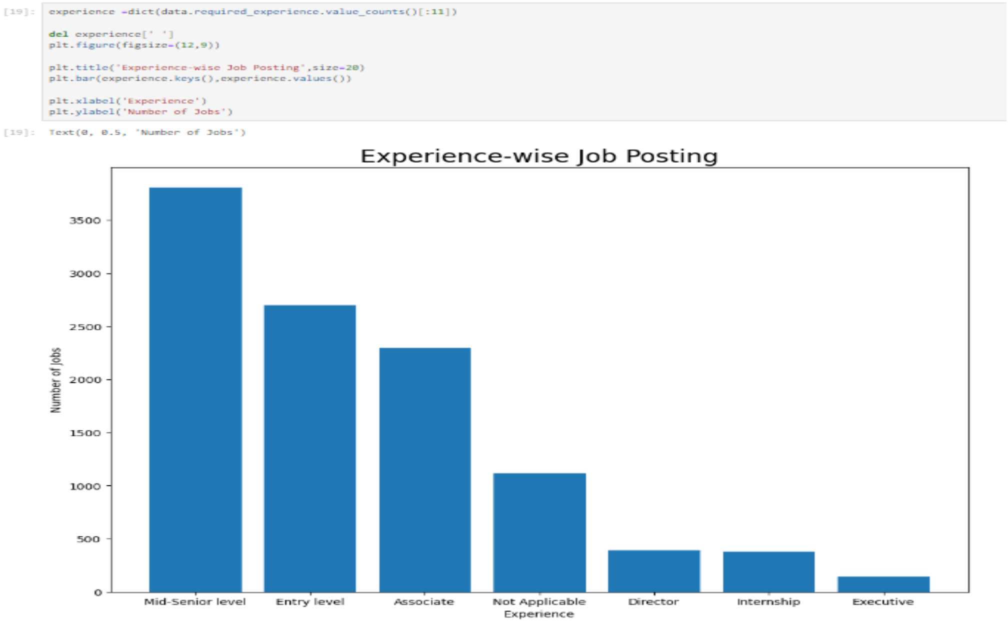


Figure 4.4 Proportion of Fraudulent Job Posting

The above figure 4.4 represents the proportion of fake job posting. This metric, known as the true positive rate or recall, indicates the model's ability to correctly classify fraudulent job postings out of all actual fraudulent postings in the dataset. A high proportion suggests that the model is effective in identifying real activity, while a lower proportion may indicate areas for improvement which is fraudulent.

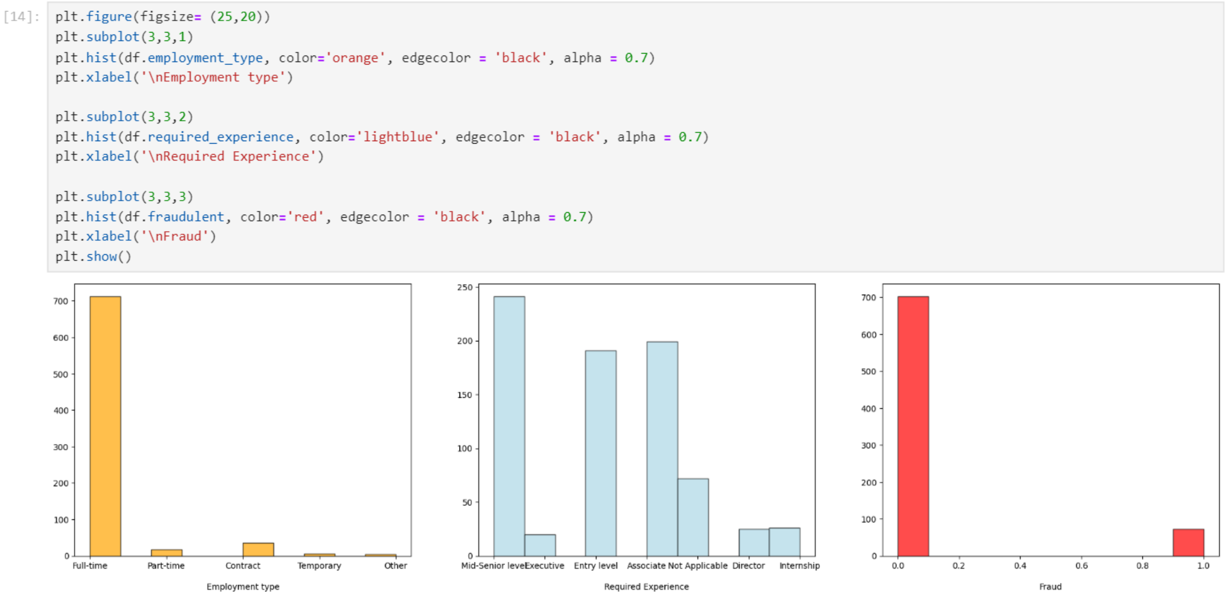
Additionally, analyzing the distribution of fraudulent job postings across different categories or features, such as industry or job location, can provide further insights into the model's effectiveness and potential areas of bias. Overall, monitoring the proportion of fraudulent job postings correctly identified by the model is crucial for evaluating its performance and ensuring its effectiveness in detecting fraudulent activity in job postings.



```
[19]: experience =dict(data.required_experience.value_counts()[:11])

      del experience[' ']
      plt.figure(figsize=(12,9))

      plt.title('Experience-wise Job Posting',size=20)
      plt.bar(experience.keys(),experience.values())

      plt.xlabel('Experience')
      plt.ylabel('Number of Jobs')
[19]: Text(0, 0.5, 'Number of Jobs')
```

Figure 4.5 Statistics of Experience wise Job posting

The above figure 4.5 represents the Statistics of Experience wise Job posting



```
[14]: plt.figure(figsize= (25,20))
      plt.subplot(3,3,1)
      plt.hist(df.employment_type, color='orange', edgecolor = 'black', alpha = 0.7)
      plt.xlabel('\nEmployment type')

      plt.subplot(3,3,2)
      plt.hist(df.required_experience, color='lightblue', edgecolor = 'black', alpha = 0.7)
      plt.xlabel('\nRequired Experience')

      plt.subplot(3,3,3)
      plt.hist(df.fraudulent, color='red', edgecolor = 'black', alpha = 0.7)
      plt.xlabel('\nFraud')
      plt.show()
```

Figure 4.6 Statistic of Employment type, Required Experience, Fraud

The above figure 4.6 represents Statistics of Employment type, Required Experience, Fraud.

4.2 NON-TECHNICAL

- **Project Planning and Coordination:** Comprehensive project plans, outlining tasks, milestones, and timelines to ensure smooth project execution and adherence to deadlines.
- **Risk Management**: Identify potential risks and uncertainties associated with the project, develop mitigation strategies, and proactively address challenges to minimize disruptions and ensure project success.
- **Ethical Considerations:** Consider ethical implications related to data collection, usage, and model interpretation, ensuring compliance with regulatory requirements and ethical standards. Prioritize fairness, transparency, and accountability in decision-making processes to uphold integrity and trustworthiness in the project outcomes.
- **Continuous Learning and Improvement:** Actively seek opportunities for professional development and learning, staying updated on industry trends, best practices, and emerging technologies in data science and predictive modeling. Apply acquired knowledge and skills to continuously improve project outcomes and deliver value to stakeholders.
- **Quality Assurance:** Implement robust quality assurance processes to ensure the accuracy, reliability, and usability of project deliverables. Conduct thorough testing and validation of the predictive model, identifying and resolving any issues or discrepancies to deliver high-quality results.
- **Time Management:** Effectively manage time and prioritize tasks to maximize productivity and meet project deadlines. Employ time management techniques such as setting goals, breaking tasks into manageable chunks, and minimizing distractions to optimize efficiency and workflow.

CHAPTER- 5

CONCLUSION

In conclusion, the data science internship has been an invaluable learning experience, providing practical insights into the application of data science concepts in real-world scenarios. Through the fraudulent job prediction project, gained hands-on experience in data preprocessing, model development, and performance evaluation using Python programming language and Jupyter Notebook environment.

Technical achievements include the implementation of essential Python libraries and tools for data manipulation, analysis, and visualization. Leveraging techniques such as word cloud visualization and performance analysis, successfully developed a predictive model using the Random Forest algorithm to detect fraudulent job postings with high accuracy.

Beyond technical skills, also honed non-technical competencies such as stakeholder engagement, project planning, ethical considerations, continuous learning, and quality assurance. Effective communication, risk management, and time management were essential in ensuring project success and delivering high-quality outcomes.

Overall, the data science internship has been a transformative journey, empowering to make meaningful contributions to the field and preparing for the opportunities and challenges that lie ahead.

REFERENCES

[1] S. Dutta and S. K. Bandyopadhyay, "Fake Job Recruitment Detection Using Machine Learning Approach," *International Journal of Engineering Trends and Technology (IJETT)*, vol. 68, no. 1, pp. 23-28, 2020.

[2] Simran, S. B. Kodli, and S. Shastri, "Prediction of Fake Job Posting Using Machine Learning," *International Research Journal of Modernization in Engineering Technology and Science*, vol. 4, no. 10, pp. 1691-1695, October 2022.

[3] P. Khandagale, A. Utekar, A. Dhonde, and Prof. S. S. Karve, "Fake Job Detection Using Machine Learning," *International Journal for Research in Applied Science & Engineering Technology (IJRASET)*, vol. 10, no. IV, pp. 1822-1825, April 2022.

[4] Gulshan P., T. Mukund, A. Ajay, P. Kumar, M. G. Aruna, and S. H. Malatesh, "Fake Job Post Prediction Using Machine Learning Algorithms," *International Journal of Innovative Research in Technology (IJIRT)*, vol. 9, no. 3, pp. 286, August 2022, ISSN: 2349-6002.

[5] S. Vidros, C. Kolias, G. Kambourakis, and L. Akoglu, "Automatic Detection of Online Recruitment Frauds: Characteristics, Methods, and a Public Dataset," *Future Internet*, vol. 9, no. 6, pp. 6, 2017.

[6] Alharby, "An Intelligent Model for Online Recruitment Fraud Detection," *Journal of Information Security*, vol. 10, pp. 155-176, 2019.

[7] D. Ranparia, S. Kumari, and A. Sahani, "Fake Job Prediction using Sequential Network," in *2020 IEEE 15th International Conference on Intelligent Systems and Control (ISCO)*, IEEE, 2020.