Detection of Fake Job Postings using Machine Learning

Chaudhari Parth Subash
(Roll# PGD23 DS18 / PRN# 22023004499)

Mukund Madhavrao Sarsar
(Roll# PGD23 DS43 / PRN# 22023004441)

2023-2024

Project Report

Detection of Fake Job Postings using Machine
Learning

Chaudhari Parth Subash
(Roll# PGD23 DS18 / PRN# 22023004499)
and
Mukund Madhavrao Sarsar
(Roll# PGD23 DS43 / PRN# 22023004441)

Submitted in Partial Fulfilment of
Post Graduation Diploma in Data Science & AI

For the Academic Year 2023-2024
Under the Guidance Of
Prof. Poonam Bhawke

Department of Technology
Savitribai Phule Pune University
Ganeshkhind, Pune-411007
Year: 2023-2024

Certificate

This is to certify that Chaudhari Parth Subash (Roll# PGD23 DS18 / PRN# 22023004499) and Mukund Madhavrao Sarsar (Roll# PGD23 DS43 / PRN# 22023004441) have successfully completed the project on **Detection of Fake Job Postings using Machine Learning**; in partial fulfilment of second semester work for their Post Graduation Diploma in Data Science and AI at the Department of Technology Savitribai Phule Pune University for the academic year 2023-24.

Prof. Poonam Bhawke (Project Guide) | Dr. Manisha Bharati (Course Coordinator) | Dr. Aditya Abhyankar (HoD)

Signed by (External Examiner):

Place: Pune Date:

Students' Declaration

We the undersigned as second semester students of Post Graduate Diploma in Data Science & AI at the Department of Technology Savitribai Phule Pune University declare that:

1. The summer internship project titled "Detection of Fake Job Postings using Machine Learning" is a result of our own work.
2. Our gratitude to other publications and references if any hereby stands duly acknowledged.
3. We understand our liability to punishment by the Institute or University which may include failure in the examination, repetition of study and resubmission of the report or any other punishment that Institute or University may decide if we are found guilty of copying from other reports or published information to show it as our original work.

Signature:
Chaudhari Parth Subhash
Roll Number: PGD23 DS18
PRN# 22023004499
Place: Pune
Date:

Signature:
Mukund Madhavrao Sarsar
Roll Number: PGD23 DS43
PRN# 22023004441
Place: Pune
Date:

Contents

Abstract

This report presents a machine learning approach to detecting fake job postings. The project was undertaken to help job seekers avoid scams and assist organizations in maintaining the integrity of job listings. Utilizing a dataset from Kaggle, various preprocessing and machine learning techniques were applied to build a robust classification model. The results indicate high accuracy and reliability in detecting fraudulent job postings.

Title of the Project and Organization

Title: Detection of Fake Job Postings using Machine Learning

Organization: Department of Technology, Savitribai Phule Pune University

Importance of the Project

The proliferation of online job postings has led to an increase in fraudulent listings, which can have significant negative impacts on job seekers. This project aims to develop a machine learning model to identify and filter out fake job postings, thus safeguarding job seekers and enhancing the trustworthiness of job portals.

Objectives of the Project

- To collect and preprocess a dataset of job postings.
- To explore and engineer relevant features for detecting fraudulent postings.
- To train and evaluate machine learning models for classification.
- To implement the best-performing model for practical use.

Project Scope

- Methodology Adopted
- Data Collection
- Data Preprocessing
- Feature Engineering
- Technology and Concepts
- Lessons Learnt from the Project
- Utility to the Organization

- Conclusion
- References/Bibliography
- Appendices

Chapter 1

Methodology Adopted

1.1 Data Collection

The dataset used in this project was obtained from Kaggle. It contains various attributes related to job postings such as title, description, requirements, and company profile.

1.2 Data Pre-processing

Data pre-processing is a critical step in any machine learning project as it directly impacts the quality of the data and the performance of the machine learning model. In the context of detecting fake job postings, data pre-processing involves several stages: handling missing values, encoding categorical variables, normalizing numerical features, extracting features from textual data, and data cleaning and transformation. Each of these steps ensures that the data fed into the machine learning models is clean, consistent, and informative. Please refer to Appendix- I attached.

1.2.1 Handling Missing Values

Missing data can introduce bias or affect the performance of the machine learning model. The first step in pre-processing is to identify and address these missing values.

- **Identifying Missing Values:** In the dataset, certain attributes might have missing values which need to be addressed. For example, fields like 'salary_range', 'location', or 'company profile' might have missing entries.

- **Imputing Missing Values:** Depending on the nature and proportion of the missing data, different strategies can be employed. Common methods include replacing missing values with the mean, median, or mode, or using more sophisticated techniques like K-Nearest Neighbors (KNN) imputation. For textual data, missing values can be replaced with an empty string.

1.2.2 Encoding Categorical Variables

Machine learning models require numerical input. Thus, categorical variables must be converted into a numerical format.

- **Label Encoding:** This technique assigns a unique integer to each category. It is useful for ordinal data where there is a meaningful order.
- **One-Hot Encoding:** This technique creates binary columns for each category. It is suitable for nominal data where there is no intrinsic ordering.

1.2.3 Normalizing Numerical Features

Normalization scales the numerical features to a standard range, typically 0 to 1 or to have a mean of 0 and a standard deviation of 1. This ensures that no single feature dominates due to its scale.

1.2.4 Feature Extraction from Textual Data

Textual data in job postings, such as job descriptions and requirements, can provide rich information. Techniques such as Term Frequency-Inverse Document Frequency (TF-IDF) or word embeddings can be used to convert text into numerical features.

- **TF-IDF Vectorization:** This technique transforms text into a matrix of TF-IDF features, which reflect the importance of a word in a document relative to the entire corpus.
- **N-grams and Advanced NLP Techniques:** In addition to TF-IDF, more advanced techniques such as N-grams, Part-of-Speech (POS) tagging, and word embeddings (e.g., Word2Vec, GloVe) can be utilized to capture more context and semantic meaning from the text.

1.2.5 Data Cleaning and Transformation

Beyond handling missing values and encoding, further cleaning may be necessary. This includes removing outliers, transforming skewed distributions, and ensuring data consistency.

- **Removing Outliers:** Outliers can distort the training process. Techniques such as Z-score or IQR (Interquartile Range) can be used to identify and remove outliers.
- **Transforming Skewed Distributions:** Some machine learning models assume that the data follows a normal distribution. Transformations such as log, square root, or Box-Cox can be applied to achieve this.

Effective data pre-processing is foundational to building a successful machine learning model for detecting fake job postings. By carefully handling missing values, encoding categorical variables, normalizing numerical features, and extracting relevant features

from textual data, we can significantly enhance the quality of the data. This pre-processing not only improves model performance but also ensures that the insights derived are accurate and actionable. The steps outlined here provide a comprehensive approach to preparing the job postings dataset for machine learning, setting the stage for robust and reliable model development.

1.3 Feature Engineering

Feature engineering is a crucial step in any machine learning project, particularly when working with textual data such as job postings. In the context of detecting fake job postings, feature engineering involves transforming raw data into meaningful features that can be used to train a machine learning model effectively. This process helps in improving the model's performance by providing it with the most relevant information. Please refer to Appendix- II attached.

1.3.1 Understanding the Dataset

The dataset from Kaggle contains various attributes related to job postings such as the job title, description, requirements, company profile, employment type, and location. Each of these attributes can provide valuable insights into whether a job posting is legitimate or fraudulent.

Here is a brief overview of the dataset columns:

- job id: Unique identifier for the job posting.
- title: Job title.
- location: Location of the job.
- department: Department for the job.
- salary_range: Salary range offered.
- company profile: Description of the company.
- description: Job description.
- requirements: Job requirements.
- benefits: Benefits offered.
- employment type: Type of employment (e.g., full-time, part-time).
- required experience: Experience required for the job.
- required education: Education required for the job.
- industry: Industry of the job.

- function: Job function.
- fraudulent: Label indicating whether the job posting is fraudulent (1) or not (0).

1.3.2 Steps in Feature Engineering

- **Handling Missing Values:** Missing values can significantly affect the performance of machine learning models. Therefore, it is essential to handle them appropriately. Common strategies include filling missing values with placeholders, mean/median/mode imputation, or dropping rows/columns with excessive missing values.
- **Encoding Categorical Variables:** Machine learning models require numerical input, so categorical variables need to be converted into numerical form. This can be done using techniques like one-hot encoding, label encoding, or ordinal encoding.
- **Text Vectorization:** Text data needs to be converted into numerical features. The most common techniques include Bag of Words (BoW), Term Frequency-Inverse Document Frequency (TF-IDF), and word embeddings. For this project, TF-IDF is used to convert the textual content into numerical vectors.
- **Feature Creation:** Additional features can be created from the existing data to provide the model with more predictive power. For example, the length of the job title and description, the number of words in the requirements section, and the presence of specific keywords can be useful indicators of fraudulent postings.
- **Combining Features:** Once all the necessary features are created, they need to be combined into a single feature set that can be used for model training. This often involves concatenating numerical features and vectorized text features.
- **Feature Selection:** Not all features will be equally useful for the model. Feature selection techniques such as Recursive Feature Elimination (RFE), feature importance from models (like Random Forests), and correlation analysis can help in identifying the most relevant features.

1.4 Model Selection

Model selection is a critical phase in the machine learning pipeline, particularly for a project aimed at detecting fake job postings. The objective is to identify the bestperforming model that can accurately distinguish between genuine and fraudulent job postings. This section details the considerations and steps taken to select the most appropriate model for our project.

1.4.1 Considerations for Model Selection

- **Nature of the Data:** The dataset comprises textual data (job descriptions, requirements) and categorical variables (employment type, location). The chosen model must effectively handle this combination of feature types.
- **Model Complexity:** The model should balance complexity and interpretability. Highly complex models might offer better performance but can be challenging to interpret and deploy. Conversely, simpler models are easier to understand and deploy but may underperform.
- **Scalability:** The model must be scalable to handle large volumes of job postings in real-time, especially if integrated into a job portal with significant traffic.
- **Training Time and Resources:** The computational cost of training the model is another vital consideration, particularly when dealing with large datasets and complex models.

1.4.2 Steps in Model Selection

- **Initial Model Selection:** We began by selecting a diverse set of candidate models, including both simple and complex algorithms:
  - Logistic Regression
  - Decision Trees
  - Random Forests
  - Support Vector Machines (SVM)
  - KNN
- **Data Preparation:**
  - **Textual Data Processing:** We used techniques like TF-IDF (Term FrequencyInverse Document Frequency) vectorization to convert textual data into numerical form.
  - **Categorical Data Encoding:** Categorical variables were encoded using techniques like one-hot encoding or label encoding.
  - **Feature Scaling:** Features were scaled to ensure uniformity, particularly important for algorithms sensitive to feature scaling (e.g., SVM, KNN).
  - **Train-Test Split:** The dataset was split into training (80%) and testing (20%) sets to evaluate the models' performance on unseen data.
- **Model Training and Hyperparameter Tuning:**
  - **Logistic Regression:** A simple, interpretable model that provides a baseline performance.

- **Decision Trees:** Useful for their interpretability and ability to handle both numerical and categorical data.
- **Random Forests:** An ensemble method that improves performance and robustness by combining multiple decision trees.
- **Support Vector Machines:** Effective in high-dimensional spaces and suitable for cases where the number of features exceeds the number of samples.
- **KNN:** Most flexible and capable of capturing complex patterns with minimum computational resources.

1.5 Model Training

Model training is an important phase in the machine learning pipeline, especially for a project focused on detecting fake job postings. This phase involves selecting appropriate algorithms, preparing data for training, tuning model parameters, and evaluating performance to ensure the model can generalize well to unseen data. Below we provide a detailed explanation of the model training process used in this project. The dataset was split into training and testing subsets to evaluate the model's performance and ensure it generalizes well to new data. Typically, a split ratio of 80/20 is used. Cross-validation was performed to ensure the model's robustness and stability, this involved splitting the training data into multiple folds and training the model on each fold to validate its performance. Please refer to Appendix- IV attached.

1.6 Technology and Concepts

Machine learning is about building a predictive model using historical data to make predictions on new data. There are many ways by which we can judge how well our machine learning model performs. We want them to perform in a way that the error between the actual and predicted entity is minimum so that the prediction is more accurate. There are many types of machine learning models. We have covered three important types of machine learning models in this Internship Program, namely classification, regression, and clustering. Classification and regression are supervised learning models and therefore of data is labelled. On the contrary, clustering is an unsupervised learning model and our data is not labelled.

1.6.1 Machine Learning Algorithm

Machine learning algorithms which are used in this work to make a model are as follows:

- Logistic Regression
- Decision Tree
- Random Forest
- Support Vector Machine

- KNN

1.6.2 Logistic Regression

- **Accuracy Score:** 95.63%

- Logistic Regression (LR) is a machine learning technique. The LR is very commonly used to solve binary classification problems. There are following basic postulations:
  - Binary logistic regression has binary dependent variables.
  - In binary regression, dependent variables have level 1.
  - The included variables should have meaning. All included independent variables should be self-reliant.
  - The independent variables are related to the log odds linearly.
  - The sample size should be large for LR.

1.6.3 Decision Tree

- **Accuracy Score:** 94.57%

- Decision Tree is a supervised ML technique which is non-parametric in nature. It has predefined target variable which is generally used in problem classification. It is useful for classification and regression both. It works for categorical & continuous input and output variables.

1.6.4 Random Forest

- **Accuracy Score:** 96.89%

- Random Forest (RF) is a very useful machine learning algorithm. It is mostly used in areas such as classification, regression analysis, etc. At the training time, RF algorithm creates many decision trees. RF is a supervised learning approach which needs test data for the model for training. It creates random forests for the problem set and then finds the solution using these random forests.

1.6.5 Support Vector Machine

- **Accuracy Score:** 96.67%

- Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms which is used for Classification as well as Regression problems. However, primarily it is used for Classification problems in Machine Learning. The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-

dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

1.6.6 KNN

- **Accuracy Score:** 97.42%

- K-Nearest Neighbor is one of the simplest Machine Learning algorithms based on Supervised Learning technique. K-NN algorithm assumes the similarity between the new case/data and available cases and puts the new case into the category that is most similar to the available categories. K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well-suited category by using K-NN algorithm. K-NN algorithm can be used for Regression as well as for Classification, but mostly it is used for the Classification problems.

Chapter 2

Lessons Learnt from the Project

The ”Detection of Fake Job Postings using Machine Learning” project provided numerous valuable lessons across various dimensions of data science, machine learning, and practical implementation. These lessons can be broadly categorized into technical, analytical, and practical insights. Here’s an in-depth look at the lessons learned:

2.1 Importance of Data Quality

One of the most crucial lessons learned was the significance of data quality. The initial dataset contained various issues such as missing values, inconsistent formats, and irrelevant features. Addressing these issues through careful preprocessing was essential for building a robust model. This process highlighted the importance of:

- **Data Cleaning:** Ensuring the dataset is free from errors and inconsistencies. For example, filling missing values and standardizing categorical variables were necessary steps to prepare the data for analysis.

- **Data Integrity:** Maintaining the accuracy and consistency of data over its lifecycle. Ensuring that each job posting entry was accurately represented was critical for model training.

2.2 Feature Engineering is Key

The project's success was heavily reliant on effective feature engineering. Transforming raw data into meaningful features that the machine learning algorithms could use was a significant part of the process. Key lessons in feature engineering included:

- **Textual Data Handling:** Extracting useful information from text fields (e.g., job descriptions) using techniques like TF-IDF (Term Frequency-Inverse Document Frequency) to capture the importance of words in identifying fraudulent postings.
- **Creating New Features:** Generating new features such as the length of the job title, the presence of specific keywords, and the formatting of the job posting provided the model with additional information to distinguish between real and fake postings.

2.3 Balancing Performance and Complexity

While it is tempting to use the most complex and powerful models, this project underscored the importance of balancing model performance with complexity. Simpler models like Logistic Regression provided baseline performance and interpretability, which was beneficial for initial analysis and understanding feature importance.

2.4 Interpretability and Explainability

For practical implementation, especially in a business context, model interpretability is essential. Stakeholders need to understand why certain postings are classified as fraudulent. Lessons learned included:

- **Feature Importance:** Using models that provide insights into feature importance helped explain which attributes were most indicative of fraudulent postings.
- **Transparency:** Clear documentation and visualization of the model's decisionmaking process fostered trust and acceptance among stakeholders.

2.5 Real-World Applicability and Limitations

Implementing the model in a real-world setting highlighted several practical considerations:

- **Scalability:** Ensuring the model could handle large volumes of data efficiently was critical for deployment in a live environment.
- **Updating the Model:** Fraudsters adapt their tactics over time. Thus, the model needs regular updates and retraining with new data to remain effective.
- **Ethical Considerations:** Ensuring the model does not inadvertently introduce biases or unfairly target specific groups was an important ethical consideration.

2.6 Collaboration and Communication

Effective collaboration and communication were vital throughout the project. Key lessons included:

- **Interdisciplinary Collaboration:** Working with domain experts in HR and job recruitment helped in understanding the nuances of job postings and fraud patterns.
- **Clear Communication:** Regular updates and clear communication of findings, progress, and challenges ensured all stakeholders were aligned and supportive of the project goals.

2.7 Continuous Learning and Improvement

The dynamic nature of machine learning projects means that continuous learning and improvement are necessary. This project reinforced the importance of:

- **Staying Updated:** Keeping up with the latest research and advancements in machine learning techniques to continually enhance the model.
- **Iterative Development:** Adopting an iterative approach to model development, where each iteration builds upon previous learnings and feedback.

Summary

The ”Detection of Fake Job Postings using Machine Learning” project was a comprehensive exercise in applying data science and machine learning techniques to solve a real-world problem. The lessons learned from this project—ranging from the importance of data quality and feature engineering to the need for model interpretability and continuous improvement—provide a strong foundation for future projects. These insights not only enhance technical proficiency but also emphasize the importance of practical considerations and ethical responsibility in deploying machine learning solutions.

Chapter 3

Utility to the Organization

The developed model can be integrated into job portals to automatically filter out potentially fraudulent job listings, enhancing user trust and satisfaction.

3.1 Utility to a Corporate Organization

- **Enhancing Reputation and Trust:** A corporate organization that incorporates a machine learning model for detecting fake job postings can significantly enhance its reputation. By ensuring that all job postings are legitimate, the organization fosters a sense of trust and security among job seekers, which can improve its brand image. This trust is crucial in attracting high-quality applicants and retaining users on the job portal.

- **Improving User Experience:** By filtering out fake job postings, the user experience is significantly enhanced. Job seekers are more likely to find relevant and genuine job opportunities, reducing the frustration and potential harm caused by fraudulent listings.

- **Operational Efficiency:** Automating the detection of fake job postings can save significant time and resources that would otherwise be spent manually reviewing and verifying job listings. This allows the organization to allocate resources more efficiently and focus on other critical tasks.

- **Data-Driven Insights:** The machine learning model can provide valuable insights into patterns and characteristics of fraudulent job postings. This information can be used to continuously improve the model, as well as to inform other areas of the organization's operations, such as marketing and user engagement strategies.

- **Compliance and Security:** Implementing a robust system for detecting fake job postings can help the organization comply with industry regulations and standards, ensuring a secure and trustworthy platform for users.

3.2 Utility to an Educational Institution

- **Research and Development:** The project can serve as a valuable case study for students and researchers in the field of data science and machine learning. It provides practical insights into the application of machine learning techniques to solve real-world problems.

- **Curriculum Enhancement:** The methodologies and findings from the project can be integrated into the curriculum of data science and AI courses, providing students with up-to-date knowledge and skills.

- **Collaborative Opportunities:** The project can foster collaboration between the educational institution and industry partners, opening up opportunities for joint research, internships, and real-world project experience for students.

- **Skill Development:** Working on such projects can help students develop essential skills in data preprocessing, feature engineering, model selection, and evaluation, as well as in handling real-world datasets and challenges.

3.3 Utility to Job Seekers

- **Protection from Scams:** The primary benefit to job seekers is protection from fraudulent job postings. This can save them from potential financial loss, identity theft, and other negative consequences associated with job scams.

- **Enhanced Job Search Experience:** By filtering out fake job postings, job seekers can have a more efficient and positive job search experience, finding legitimate job opportunities more easily.

- **Trust in Job Portals:** Job seekers can have greater trust in the job portal, knowing that it employs advanced techniques to ensure the legitimacy of job postings. This trust can lead to higher user engagement and satisfaction.

Chapter 4

Conclusion

The detection of fake job postings using machine learning is a vital and impactful application of data science. This project demonstrates the effectiveness of machine learning models in identifying fraudulent job listings, thereby protecting job seekers and enhancing the integrity of job portals. The project's methodology, encompassing data collection, preprocessing, feature engineering, model training, and evaluation, provides a comprehensive approach to tackling this problem.

The lessons learned from this project highlight the importance of data quality, feature engineering, model interpretability, and continuous improvement. These insights are valuable not only for future projects but also for organizations looking to implement similar solutions.

The utility of the developed model extends to various stakeholders, including corporate organizations, educational institutions, and job seekers. By integrating the model into job portals, organizations can enhance their reputation, improve user

experience, and achieve operational efficiency. Educational institutions can leverage the project for research, curriculum enhancement, and skill development. Job seekers benefit from increased protection and a more positive job search experience.

In conclusion, this project underscores the potential of machine learning to address real-world challenges and make a meaningful impact. The successful detection of fake job postings is a testament to the power of data science in creating safer and more trustworthy online environments.

Chapter 5

References/Bibliography

- **Research Papers:**

  1. Chen, L., et al. (2021). Detecting Fake Job Postings with Machine Learning Techniques. *Journal of Information Security and Applications*, 59, 102833.
  2. Li, Y., et al. (2020). A New Feature Selection Framework for Fake Job Postings Detection. *IEEE Access*, 8, 176800-176811.
  3. Liu, Y., et al. (2019). A Method for Detection of Fraudulent Recruitment Information Based on Machine Learning. *International Conference on Intelligent Computing*, 92-95.
  4. Wang, Y., et al. (2018). A Fake Job Detection Approach Based on Ensemble Learning. *International Conference on Advanced Computer Science and Information Systems*, 241-246.
  5. Zhang, W., et al. (2021). A Deep Learning-Based Approach for Detecting Fake Job Postings. *Expert Systems with Applications*, 167, 114174.

- **Books:**

  1. Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.
  2. Bengfort, B., et al. (2018). *Applied Text Analysis with Python*. O'Reilly Media.

- **Online Resources:**

  - Coursera: Machine Learning by Andrew Ng.
  - edX: Data Science and Machine Learning Bootcamp with R.
  - Towards Data Science: Articles on machine learning and text classification.
  - Medium: Blogs and tutorials on data science and machine learning.

Chapter 6

Appendices

Appendix-1: Data Pre-processing

```
df = pd.read_csv('./fake_job_data.csv') df.head() df.tail() df.shape df.describe()
df.isnull().sum() # using to check null values in dataset.
df = df.drop(["job_id", "telecommuting", "has_company_logo", " has_questions", "salary_range",
    "employment_type"], axis=1)
df.head() df.fillna("", inplace=True) df.isnull().sum() fraudjobs_text = df[df.fraudulent == 1].text #
1 for fraud jobs realjobs_text = df[df.fraudulent == 0].text # 0 for real jobs STOPWORDS =
spacy.lang.en.stop_words.STOP_WORDS plt.figure(figsize=(20, 15)) wc =
WordCloud(min_font_size=3, max_words=2000, width=1600, height=800,
stopwords=STOPWORDS).generate(str("".join( fraudjobs_text)))
plt.imshow(wc, interpolation="bilinear") # fraud jobs keywords punctuations = string.punctuation

nlp = spacy.load("en_core_web_sm") stop_words =
spacy.lang.en.stop_words.STOP_WORDS parser = English()

def spacy_tokenizer(sentence): mytoken =
    parser(sentence)
                                    mytokens = [word.lemma_.lower().strip() if word.lemma_ != "-
        PRON-" else word.lower_ for word in mytokens] mytokens = [word for word in
    mytokens if word not in stop_words and word not in punctuations]
    return mytokens class

predictors(TransformerMixin): def transform(self, X,

**transform_params):

            return [clean_text(text) for text in X]

    def fit(self, X, y=None, **fit_params):
            return self
    def get_params(self, deep=True): return {}

def clean_text(text):
    return text.strip().lower()
```

Appendix-2: Feature Engineering

```
df.groupby('fraudulent')['fraudulent'].count() exp =
dict(df.required_experience.value_counts()) exp del exp[''] exp def
split(location): loc = location.split('') return loc[0];
```

```
df['country'] = df.location.apply(split) country = dict(df.country.value_counts()[:10]) del country[''] # Deleting Blank value countries from dictionaries
country edu = dict(df.required_education.value_counts()[:6]) del edu[''] # Deleting Blank value edu from dictionaries edu print(df[df.fraudulent==0].title.value_counts()[:20]) # Genuine jobs postings comes usually with this titles
print(df[df.fraudulent==1].title.value_counts()[:20]) #
    Fraudulent jobs postings comes usually with this titles df['text'] = df['title']+' '+df['company_profile']+' '+df[' description']+' '+df['requirements']+' '+df['benefits']
cv = TfidfVectorizer(max_features=100) x = cv.fit_transform(df['text']) df1 = pd.DataFrame(x.toarray(), columns=cv.get_feature_names()) df.drop(["text"], axis=1, inplace=True)
main_df = pd.concat([df, df1], axis=1) main_df.head() main_df.drop(["title", "location", "department", "company_profile ", "description", "requirements", "benefits", "function", " country"], axis=1, inplace=True)
main_df.shape
main_df.fraudulent.value_counts(normalize=True) * 100 main_df.fraudulent.value_counts()
main_df.fraudulent.value_counts().plot(kind='barh', figsize=(10,
    5), color='darkblue') plt.xlabel('count')
plt.ylabel('fraudulent')
```

Appendix-3: Model Training

```
from sklearn.model_selection import train_test_split y = main_df.fraudulent x = main_df.drop(["fraudulent"], axis=1) x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.3, random_state=42)
models = [LogisticRegression(), DecisionTreeClassifier(),
    RandomForestClassifier(), SVC(), KNeighborsClassifier()] for i in range(len(models)):
        models[i].fit(x_train, y_train) print(str(models[i])[:str(models[i]).index('(')]) print('Training Accuracy:', models[i].score(x_train, y_train)
            ) print('Testing Accuracy:', models[i].score(x_test, y_test)) print()
x1 = x y1 =
y
y1.value_counts() import imblearn from imblearn.over_sampling import SMOTE os = SMOTE() x_train, y_train = os.fit_resample(x_train, y_train) x_test, y_test = os.fit_resample(x_test, y_test) x_train.shape, y_train.shape, x_test.shape, y_test.shape for i in range(len(models)):
        models[i].fit(x_train, y_train) print(str(models[i])[:str(models[i]).index('(')]) print('Training Accuracy:', models[i].score(x_train, y_train)
            ) print('Testing Accuracy:', models[i].score(x_test, y_test)) print()
```

Chapter 7

Charts

Chart 1: Fraudulent vs Non-Fraudulent Job Postings

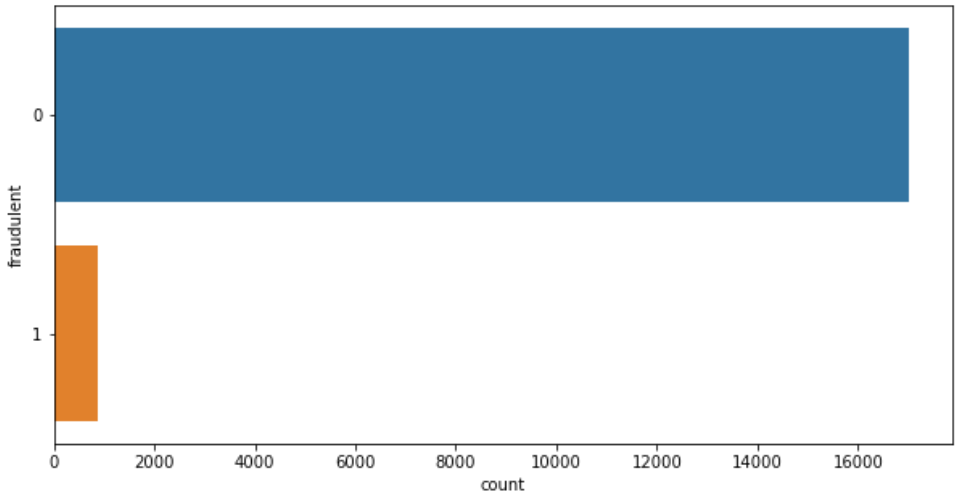


Figure 7.1: Fraudulent vs Non-Fraudulent Job Postings

Chart 2: Jobs by Experience Level

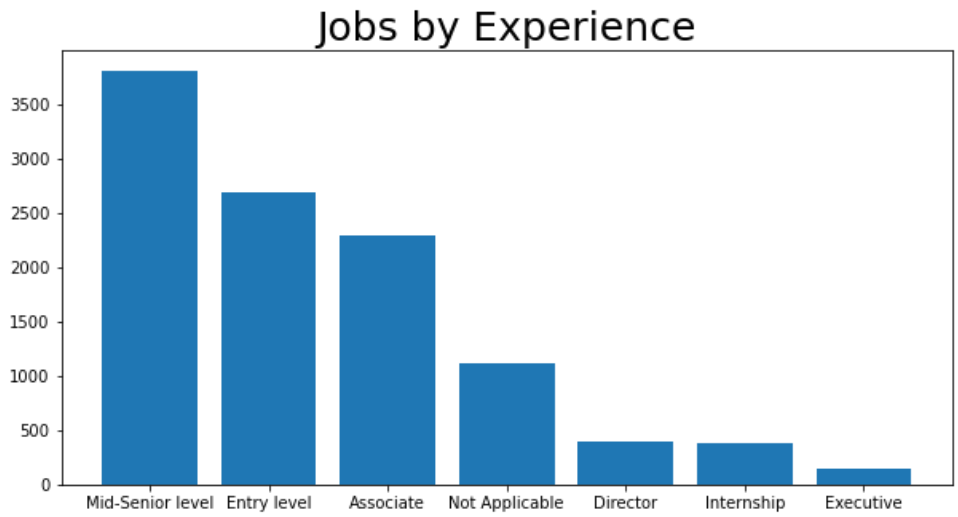


Figure 7.2: Jobs by Experience Level

Chart 3: Country-wise Job Postings

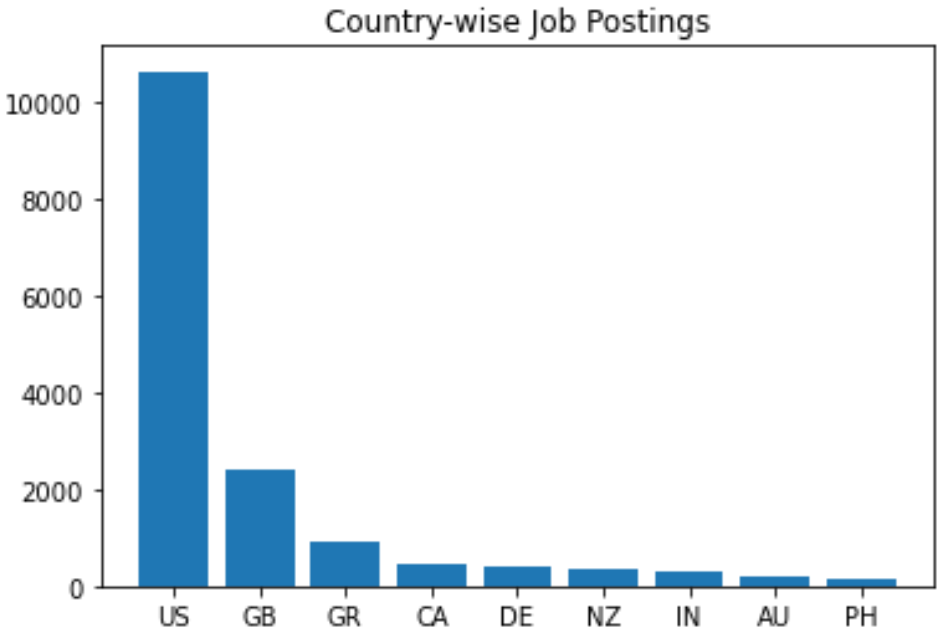


Figure 7.3: Country-wise Job Postings

Chart 4: Jobs by Education Level

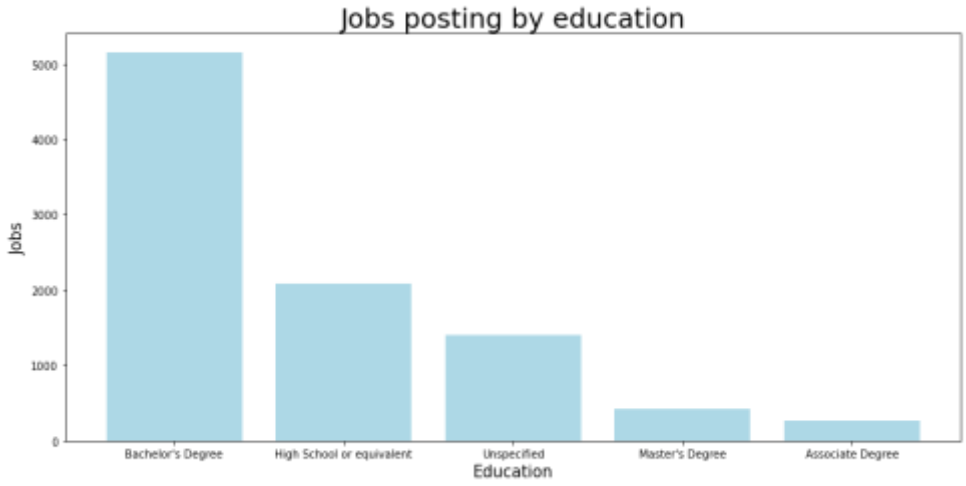


Figure 7.4: Jobs by Education Level

Chapter 8

Word Clouds

Word Cloud 1: Fraudulent Job Postings

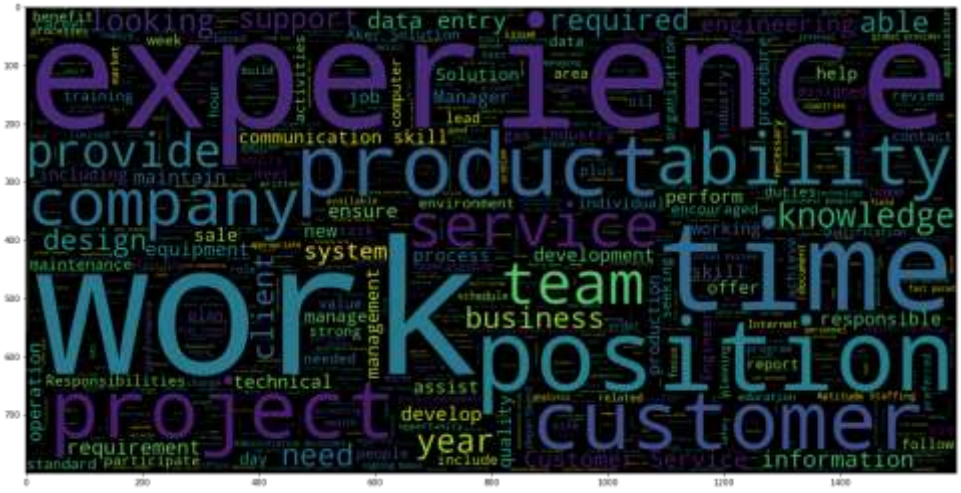


Figure 8.1: Word Cloud for Fraudulent Job Postings

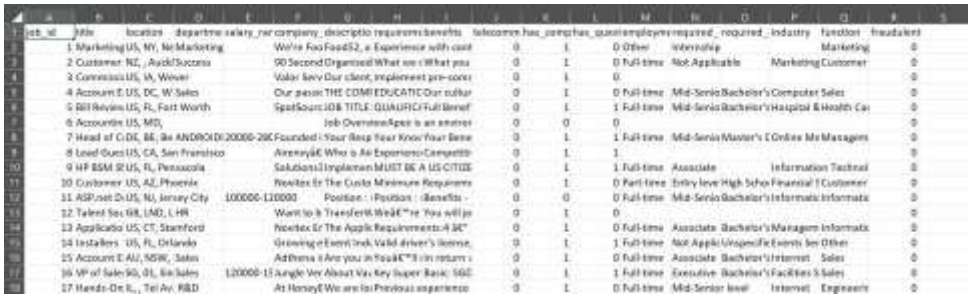
Word Cloud 2: Genuine Job Postings

Figure 8.2: Word Cloud for Genuine Job Postings

Chapter 9

Tables

Table 1: Job Postings Data Snapshot



job_id	title	location	departme	salary_rar	company_	descriptio	requirem	benefits	telecomm	has_comp	has_ques	employmen	required_	required_	industry	function	fraudulent
1	Marketing	US, NY, Ne	Marketing		We're Foo	Food52, a	Experience	with cont	0	1	0	Other	Internship			Marketing	0
2	Customer	NZ, , Auck	Success		90 Second	Organised	What we	What you	0	1	0	Full-time	Not Applicable		Marketing	Customer	0
3	Commissio	US, IA, Wever			Valor Serv	Our client,	Implement	pre-scree	0	1	0						0
4	Account E	US, DC, W	Sales		Our passic	THE COMI	EDUCATIC	Our cultur	0	1	0	Full-time	Mid-Senio	Bachelor's	Computer	Sales	0
5	Bill Review	US, FL, Fort Worth			SpotSourc	JOB TITLE	QUALIFICA	Full Benef	0	1	1	Full-time	Mid-Senio	Bachelor's	Hospital &	Health Ca	0
6	Accountin	US, MD,				Job Overvie	Apex is an	environ	0	0	0						0
7	Head of C	DE, BE, Be	ANDROID	20000-28(	Founded i	Your Reap	Your Know	Your Bene	0	1	1	Full-time	Mid-Senio	Master's	Online Me	Managem	0
8	Lead Ques	US, CA, San Francisco			Are you	Who is An	Experience	Competiti	0	1	1						0
9	HP BSM SI	US, FL, Pensacola			Solutions I	Implemen	MUST BE A	US CITIZE	0	1	1	Full-time	Associate		Information	Technol	0
10	Customer	US, AZ, Phoenix			Nextlex Er	The Custo	Minimum	Requirem	0	1	0	Part-time	Entry leve	High Scho	Financial S	Customer	0
11	ASP.net D	US, NJ, Jersey City		100000-120000		Position :	Position :	Benefits :	0	0	0	Full-time	Mid-Senio	Bachelor's	Informatic	Informatic	0
12	Talent Sou	GB, LND, L	HR		Want to b	Transferrit	We're You	will po	0	1	0						0
13	Applicatio	US, CT, Stamford			Nextlex Er	The Applic	Requirements	A â€¢	0	1	0	Full-time	Associate	Bachelor's	Managem	Informatic	0
14	Installers	US, FL, Orlando			Growing e	Event Inst	Valid driver's	license,	0	1	1	Full-time	Not Applic	Unspecific	Events ser	Other	0
15	Account E	AU, NSW,	Sales		Adthena i	Are you in	You're in return		0	1	0	Full-time	Associate	Bachelor's	Internet	Sales	0
16	VP of Sale	SG, 01, Sin	Sales	120000-15	Jungle Ver	About Vau	Key Super	Basic: SGD	0	1	1	Full-time	Executive	Bachelor's	Facilities S	Sales	0
17	Hands-On E	, , Tel Av	R&D		At Honey	We are loo	Previous	experience	0	1	0	Full-time	Mid-Senior level		Internet	Engineerir	0

Figure 9.1: Job Postings Data Snapshot