Detecting Online Recruitment Fraud by using Machine Learning

by

Gitanjali Ghosh
17101228
Hridita Tabassum
17101446
Afra Atika
17101206
Zainab Kutubuddin
17101198

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
BRAC University
January 2021

Declaration

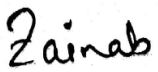
1. The thesis submitted is our own original work while completing degree at BRAC University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Gitanjali Ghosh
17101228

Hridita Tabassum
17101446

Afra Atika
17101206



Zainab Kutubuddin
17101198

Approval

The thesis titled "Detecting Online Recruitment Fraud Using Machine Learning" submitted by

1. Gitanjali Ghosh (17101228)
2. Hridita Tabassum (17101446)
3. Afra Atika (17101206)
4. Zainab Kutubuddin (17101198)

Of Fall, 2020 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science.

Examining Committee:

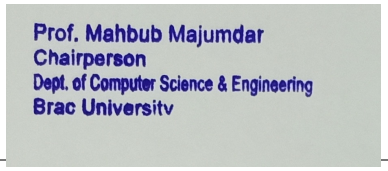
Supervisor:
(Member)

Dr. Amitabha Chakrabarty
Associate Professor
Department of Computer Science and Engineering
BRAC University

Program Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam
Associate Professor
Department of Computer Science and Engineering
BRAC University

Head of Department:
(Chair)



Dr. Mahbubul Alam Majumdar
Professor and Chairperson
Department of Computer Science and Engineering
BRAC University

Dedication

We would like to dedicate our thesis paper to our parents and family for their continuous support throughout our research. Special gratitude towards our friends for always being by our side.

Acknowledgement

Firstly, all praise to the Almighty Creator for whom our thesis have been completed without any major interruption.
Secondly, to our supervisor Dr. Amitabha Chakrabarty sir for his kind support and advice in our work. He guided us whenever we needed help.
And finally to our parents whose constant support have paved way for us to make this research successful. With their kind support and prayer we are now on the verge of our graduation.

Abstract

Online Recruitment fraud (ORF) is becoming an important issue in the cyber-crime region. Companies find it easier to hire people with the help of the internet rather than the old traditional way. But it has greatly attracted the scammers to deceive people and exploit their information. There have been lots of incidents where innocent people have fallen for this malicious fraud and lost millions of money. Even it causes harm to business and the economy. Unlike other cyber-security problems, like email spam, phishing, opinion fraud, detecting Online Recruitment Fraud(ORF) did not get that much of recognition. So, this matter needed to be highlighted more. In this paper, we have proposed a solution on how to detect ORF. We have presented our results based on the previous model and also presented the methodologies which we are going to use to create the ORF detection model where we are using our own dataset. We are going to use a publicly accessible dataset from fake_job_postings.csv, license-CC0: Public Domain, as a reference for the dataset that we have created. Furthermore, we have collected 4000 data from different job sites in Bangladesh, among which 301 of them are fraudulent. We have used many common and latest classification models to detect which algorithm works best for our model. Logistic Regression, AdaBoost, Decision Tree Classifier, Random Forest, Voting Classifier, LightGBM, Gradient Boosting are the algorithms that have been used. From our observations we have found that the accuracy of different prediction models are: Logistic Regression(94.67%), AdaBoost(95%), Decision Tree Classifier(95%), Random Forest(95%), Voting Classifier(95.34%), LightGBM(95.17%), Gradient Boosting(95.17%). Through this report, we tried to create a precise way for detecting the fraudulent hiring posts.

Keywords: Machine Learning; Fraud Detection; Prediction; Decision Tree Classifier; Logistic Regression algorithm; Adaptive Boosting; Random Forest Classifier; Decision trees; Gradient Boost; LightGBM

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

As technology is advancing, most of our conventional ways of working are also changing. So, when it comes to recruiting employees, it is the same here too. Companies tend to receive 1000+ resumes each week and it is quite hard and time consuming to go through each one of them individually. Hence corporations are moving the critical procedure of hiring through clouds. The Online Recruitment process is helpful for both the recruiters and job seekers. Most companies are adopting the ATS software for hiring workers. ATS (Applicant Tracking System) is a software application which enables organizations to hire workers and job seekers to find jobs effectively and precisely. Usually, for hiring employees, companies need to publish advertisements, collect resumes, select the perfect candidates. All this requires a lot of time but using ATS all these can be done over a short span of time. In the beginning ATS system were merely used by big businesses but now it is being adopted by small firms too. In fact, an article stated that, in 2015 there were 5 million job openings in the USA and most of them were posted online [1]. But this has taken the attention of scammers who created a new field for fraudulent activities. Generally, the fraudsters try to give thriving offers in order to grab the attention of the candidates and later steal their personal and valuable information. Not only the candidate, but also the companies are affected by this too, as it hampers their reputation in the mind of the job seekers. In 2015, there was a survey done by FlexJobs, which stated that, for each legitimate job advertisement there are around 60-70 job scams. Where only 48% of the candidates had the knowledge of online recruitment scams [2]. Verifying Online Recruitment fraud is an essential issue which requires more awareness but only a limited amount of work has been done in this sector. In this work, a model is proposed which allows us to distinguish between legitimate posts and fraudulent posts, by using few classification algorithms, which are described in depth later in the report. For analyzing the proposed model, we are going to use a dataset that we have created.

1.2 Problem Statement

Bangladesh is a developing country with an aim to transform into a country of digital economy by 2021 and a knowledge-based economy by 2041 [3]. Every sector in our country aims towards digitalization. But with it also comes some drawbacks. Nowadays, the online job recruitment process is taking a new shape. Companies post advertisements for jobs and employees apply online for these job posts in online recruitment sites. This online process of hiring new employees is proved to be more time efficient and less hectic both for the companies as well as the job seekers. But as said earlier, these processes also bring some drawbacks with it. Recently, incidents of various crimes regarding online recruitment have been reported. Due to this process, a lot of people had to suffer from personal information leakage. Scammers tend to publish ads with lucrative offers which often take people's attention and draw them towards those fabricated posts. Generally people tend to fall under this trap as they do not have enough knowledge about this issue. Such incidents happened where financial losses, confidential information such as bank account number etc. were leaked [4]. Also, the scammers are becoming really good at creating these ads so it is becoming quite difficult for people to distinguish between fake and real posts. Furthermore, in some extreme cases, sometimes people are called for interviews through these ads and later on they have to face serious consequences. For all these above reasons it has become a necessity to enquire whether a company's job advertisement is real or it is a scam. Till now, very few research has been done to build a system that would detect online recruitment fraud. In this report, we aim to research extensively on this field and build a system that would overcome this problem.

1.3 Research Objective

Today we are living in a digital world and the main tool which this digital world uses is the internet. From banking to recharging mobile phone's balance each and everything is done by the internet now-a-day. In the same way, the internet is considered to be the best platform both for applying and seeking jobs. Now people do not need to physically go from one office to another for collecting job circular or to apply for a job. With the use of the internet by sitting in their own home people can easily search for new jobs and apply within a few seconds. But as there are 2 sides of every coin, there are also some negative sides in this online employee recruitment system. Cybercrime is the most threatening and dangerous one. Based on cyber security ventures report 2021, the cost of cybercrime damages in the world is around $6 trillion annually [5]. In 2012, the Australian Bureau of Statistics published a report about personal fraud stating that 6 million people were exposed to several forms of scam, including employment scams, during any given year [6]. In cybercrime one of the most common issues is that frauds give ads about fake jobs with more salary, flexible time and other facilities which attract the job seekers, this condition is called Online Recruitment Fraud (ORF). Because of this ORF job seekers are facing a lot of troubles like leak of personal information, security problems, a well

reputed company loses its reputation and confidential information etc. which we have discussed in our problem statement. Because of all this serious problem which many job seekers are facing we want to create a model which will detect whether a particular job ad on the internet is legitimate or a hoax. By distinguishing this thing people won't fall in those traps of fake jobs anymore. Therefore, the financial problem which was created by this platform will be solved and those people who were wasting their time on those scams can be alert now. Our goal is to present a model which will give accurate results in verifying the legitimacy of recruitment ads and to eliminate the ORF issues.

1.4 Thesis Outline

The rest of the thesis paper has been organized in the following way:

Chapter 2 has the literature review of other papers that are related to our thesis work. It gives an idea of what research has been done so far.

Chapter 3 contains the methodology that we have used to detect fraud in online jobs and our work flow to conduct the research.

Chapter 4 provides detailed description of the data that we have created to perform the research and also the data preprocessing techniques that we have used to make our dataset perform better on the models.

Chapter 5 contains the detailed description of the algorithms that we have used to get better results and also provided the idea of how they work.

Chapter 6 contains the result analysis of our research. It contains feature review, data visualization and also the accuracy comparison of all the algorithms that we have used.

Chapter 7 contain the conclusion of what we have done in our research and also what we plan to do in the future to make our model better.

Chapter 2

Literature Review

As we already know that there are a huge number of fraudulent online recruitment sites who tend to exploit personal information of users and this has been a matter of concern. In different fields, there are ample amounts of study related to cybercrime detection models. Large number of papers presented ways to distinguish between malicious job opportunities and real employment advertisements, so that job seekers can avoid those scams. Whereas some introduced various indirect methods to detect the ORF, which includes Email Spam Filtering, anti-fishing technique, in order to verify the legitimacy of a post or review. An example [7], the study explains how The Online Recruitment System is beneficial for both the job seekers and the recruiters. It shows how scammers give offers and lure the candidates. They established an ensemble-based model ORF Detector for detecting fake job posts. Initially they have taken 3 classifiers, J48, Logistic Regression (LR), and Random Forest (RF). Then to build the framework they have used 3 ensembled techniques Average Vote (AV), Majority Vote (MV) and Maximum Vote (MXV) on those classifiers. The model proves to be quite beneficial as it provides an accuracy of 95.4% and average f1-score of 94%. The Logistic Regression (LR) generates a value from the data. If it is less than 0.5, then it is considered to be legitimate. Using the J48 algorithm a decision tree is made using information of each attribute. In the training phase a number of decision trees are generated using the Random Forest algorithm. During the time of prediction, most of the trees are taken in. To improve the accuracy, these are then combined with the ensembled technique, which are then compared with some threshold values. To establish the proposed framework, they have used publicly shared dataset and then the features are divided into 3 categories, Linguistic, Contextual and Metadata. Additionally, they have also presented experimental details, where they have evaluated in terms of accuracy, precision, recall etc.

Fraud in online recruitment processes can financially damage individuals and organizations. To prevent that in this paper [8] the authors proposed an intelligent model for detecting online recruitment fraud which will predict the validity of a job post by using data mining methods. In this model the two very efficient random tree method and support machine method (SVM) and an open dataset Employment Scam Aegean Dataset (EMSCAD) was used. At first, the data set was preprocessed by extracting noise and uninformative character. Then, from the labelled data of

the previous step, appropriate features were selected and extracted by using the SVM method which is a supervised learning algorithm. After that, the data was classified by using random forest ensemble classifiers. The result showed 97.41% accuracy. However, in this paper the authors used the data that is more balanced. So, the result of this method using practical data is still unknown.

In another paper [9] they proposed a model for preventing fraud in the employee recruitment process with the help of a decision support system. Decision Support System is a system that compares employee's qualifications and the job criteria and gets a result based on this decision. For this purpose, they used the Profile Matching method which works by comparing employee's competence and the job specification and assigns a value. The recruitment decision is based on this value. If the gap between the actual profile value and the expected profile value is less, then that employee is more qualified for the job. This whole process works in steps. This whole process with the help of a decision support system is said to prevent fraud in employee recruitment.

Furthermore, in another article [10] the authors said 7% job seekers are the victim of employment scams at least once in their life. They did an investigation on 17,000 job ads and found that 95% are real and 5% are fraud cases. In a statistic they found 75% of large companies use ATS (Applicant Tracking Systems) for job advertisement. Even small companies now-a-day also use ATS for hiring employees. They have said that the frauds use ATS to advertise their fake job to pretend legitimate employers. When they are successful to gain trust from the job seeker, they collect very important and confidential information from them. They mentioned that Natural Language Processing (NLP) and machine learning are used to classify textual, structural and contextual characteristics to identify the difference between real job ads and fake job ads. Regression trees, Support Vector Machines (SVM), Random Forests (RF), and neural networks are also used for scam detection.

The operation of hiring consists of a number of steps, starting from advertisement till hiring. With advancement, the critical way of recruitment has been transferred to cloud. As the traditional way of hiring requires a lot of time and energy, so the businesses are adopting the ATS (Applicant Tracking System) software. This system proved to be quite efficient as it makes the hiring process faster with maximum results. But with the increase of the ATS system, Online recruitment hoaxes are also growing. In the paper [4], it has been clearly shown how ORF problems are evolving and it tends to harm both the candidates and the business's reputation. The authors also mentioned that the ORF characteristics are reasonably similar to other cyber fraud activities like Wikipedia vandalism, cyber bullying, trolling and opinion fraud. However, where the ATS registrations are restricted, fraud verification is not required as it contacts the candidates personally. To work on their proposed program, they have used The Employment Scam Aegean Dataset (EMSCAD). To determine a fraudulent program, they have considered four fields: String, HTML fragment, Binary and Nominal. For analyzing, they have used six WEKA classifiers. They are: ZeroR, OneR, Naives Bayes, J48 decision trees, random forest and logistic regression (LR). Before giving the data to these classifiers, they apply stopword filtering. Random Forest gives the most accurate results (.914). Naive Bayes and J48 decision trees have 0.863 and 0.848 precision respectively.

Moreover, in a different research paper [11] the authors collected a report from Australian Competition and Consumer Commission (ACCC) where it is said that in October,2017 around 212,784 Australian dollars was wasted only because of 237 employment scams. Where existing methodologies used only textual and structural there they used contextual information like history of that organization, internet footprint to solve this ORF issue. Some popular job advertisements like SEEK1 suggest job seekers apply for a job before getting information about that organization's footprint. In their research they used the EMSCAD dataset which contained 17,800 real-life job ads but there was any name of the companies. therefore, they solve this problem by using two steps. Firstly, from the company profile which is listed in that dataset they extracted the names by using natural language processing toolkit and secondly, by google search they extract some contextual information of those companies. At the end for using contextual information their accuracy level was 94.29.

Furthermore, in a different article [12] authors gave different kinds of reviews for different kinds of machine learning algorithms from filtering spam. They had focused on both textual and image-based approaches. For textual detection they suggested different types of classifiers like: Naive Bayes, Support Vector Machines (SVM), Artificial Neural Networks, Logistic regression, Lazy learning, Artificial Immune Systems, Boosting, ensembles and related approaches, Hybrid methods. Among them they said SVM and RVM are faster than others. For image-based approach they suggested OCR (Optical Character Recognition), SVM, Linear classifier.

In addition, another article the authors .[13] said that they used a dataset from University of Arizona's Dark Web Project and have 192 samples to do their thesis. They tested different types of classifiers which are : Bayes models, logistic regression, classification trees, boosting, and support vector machines (SVM) to classify the forum posts. By using those they created curves of ROC(Receiver Operating Characteristic) and found that under AUC curve Svm was 89%, Naive Bayes model was 63%. Therefore, with the help of performance metrics they consider SVM and logit as best models as they detect VE recruitment with highest accuracy level which is 0.85¡ mean AUC.

In present days, online reviews play a vital role in making decisions regarding a specific product. A positive review regarding a product can increase the sales of the product whereas a negative review can decrease one. For this reason, authentic reviews are very crucial in decision making for a customer. But fake reviews can make the customer buy a not-so-good product. Therefore, it is important to identify whether a review is genuine or fake. In their paper Shivagangadhar K, Sagar H, Sathyan S, Vanipriya C.H [14] builds a system to detect fraud in online reviews using Machine Learning Techniques. They have used Naïve Bayes, Logistic Regression and Support Vector Machine classifiers for this purpose. The methods use the reviewer's profile, review data and activity of the reviewer on the internet to determine the genuineness of the review. The model is built using linguistic features like unigram presence, unigram frequency, bigram presence, bigram frequency and review length to find fake reviews. Their model requires both linguistic and behavioral features in detecting fraud in online reviews.

In today's time, spam is an important issue that needs proper attention. Every day, many new unwanted emails or messages are received for the purpose of advertisements, phishing, spreading malware etc. these mails or messages are frustrating. Email spam is the most common type of spam. Other than email spams, other spams such as blog spam, splogs are also becoming problematic. To combat spam, content analysis is necessary. In their paper, Sculley D and Wachman G M [15] uses content analysis for spam filtering. In content analysis, the actual message text is analyzed using Machine Learning Techniques. Firstly, they show that SVM classifiers actually perform worse than other classifiers. That is why they build a simple Online SVM algorithm that performs good on filtering email spams, blog comment spam and splogs. However, it is not cost efficient for large scale applications. Thus they propose Relaxed Online SVM (ROSVM) classifier for spam filtering. It is to note that the success of this method depends on the nature of spam and it is subject to change.

Chapter 3

Methodology and Work Flow

3.1 Methodology

In any research related to machine learning the first and foremost thing is to collect data. Authenticity of the dataset in a research is very important for further analysis. If the dataset is not authentic then the result of the model will not be useful at all for that research. For this data collecting we have visited a number of authentic websites for finding relevant datas like BD jobs, job.com.bd, bdjobstoday, deshijob etc. We have collected 4000 data manually and made our dataset with those data.

The next step we have done after data collection is data processing. We collected data which is understandable for humans but if we try to use those data in algorithms that it will fail to give a correct output. Therefore, for making understandable algorithms and to get higher accurate output we pre-processed our dataset in a different form which has been described later.

Figure 3.1: Steps to Predictive Learning [16]

The third thing we have done is identifying the key features from the whole dataset. We have worked with key features as they influence much to get the desirable output. Irrelevant information may case disturbance for our research so we have filtered our dataset where we dropped some irrelevant columns. The techniques that we have used here for filtering our dataset is data imputation, handling outliers through visualization, scaling the data. Visualizing the data is a very simple and common

but effective way to identify the key features of a dataset. We have used heatmap, axes plot, pie chart etc. for visualization.

After feature engineering, we have split the dataset into two parts- training set and testing set. The training dataset is used by the machine learning algorithm to learn the model. Once that is done, the testing dataset is used to predict the outcome of the model. Different machine learning algorithms and classifiers are used for this process like voting classifier, decision tree classifier, random forest, gradient boosting etc. and the accuracy of each of the algorithms are checked.

After getting all the outputs of classifiers we have noticed which classifiers work better for our dataset and which did not give that much efficient result. As we worked on our own dataset which is different from our reference dataset, we also have looked for the accuracy of our reference dataset. At the end we analyse the differences between both of the dataset's accuracy and find out why those differences have occured.

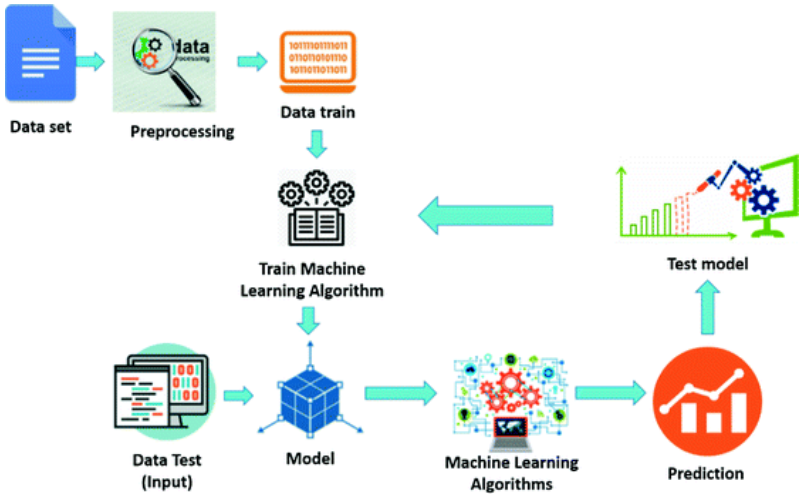


Figure 3.2: Machine Learning Predictive Model [17]

3.2 Work Flow

The first stage of our workflow was choosing a topic. After that, we have studied machine learning and data analysis. We have gathered knowledge about the existing method for predictive analysis.For that reason we go through a number of researches related to our topic. After that, we have learned about pre-processing data which is a very crucial part of machine learning. In addition to that, we have become familiarized with many features extracting and data scaling techniques. Also, we have studied the implementation of different classifiers. After acquiring all the information regarding data analysis, we have moved to our second stage.

The second stage is collecting data. We have collected many job postings from various online job portals in Bangladesh by ourselves. We have taken reference from a similar dataset "fake job postings.csv", license-CC0: Public Domain, to create our

dataset that is based on the online job postings in Bangladesh. When enough data is collected we start to process the data. We have used many visualizations to find out which features differentiate between the fake job post and real job post. Then, according to those selected features, we have categorized our data. Also, We have removed unnecessary information from the dataset in the pre-processing part. After that, we have scaled and split our data. We have used various scaling methods like MinMax, normalizer to find which method gives the most proficiency.

Finally, we have used a number of classifiers to complete our model. We have found many classifiers that may be used for detecting fraud or scams. We researched other similar papers and found some algorithms such as SVM (Support Vector Machine), decision tree classifier, random forest, NB (Naive Bayes), MLP (Multi-Level Perception), LR (Logistic Regression) etc. We have used some of the classifiers from these. We also have used some other classifiers like voting classifier, gradient boosting, LightGBM(Light Gradient Boosting) Classifier which we haven't seen in other's research. We have chosen a total of seven classifiers to make our model. Lastly, we have calculated the accuracy of these classifiers to evaluate the performance of our model and analyse the differences.

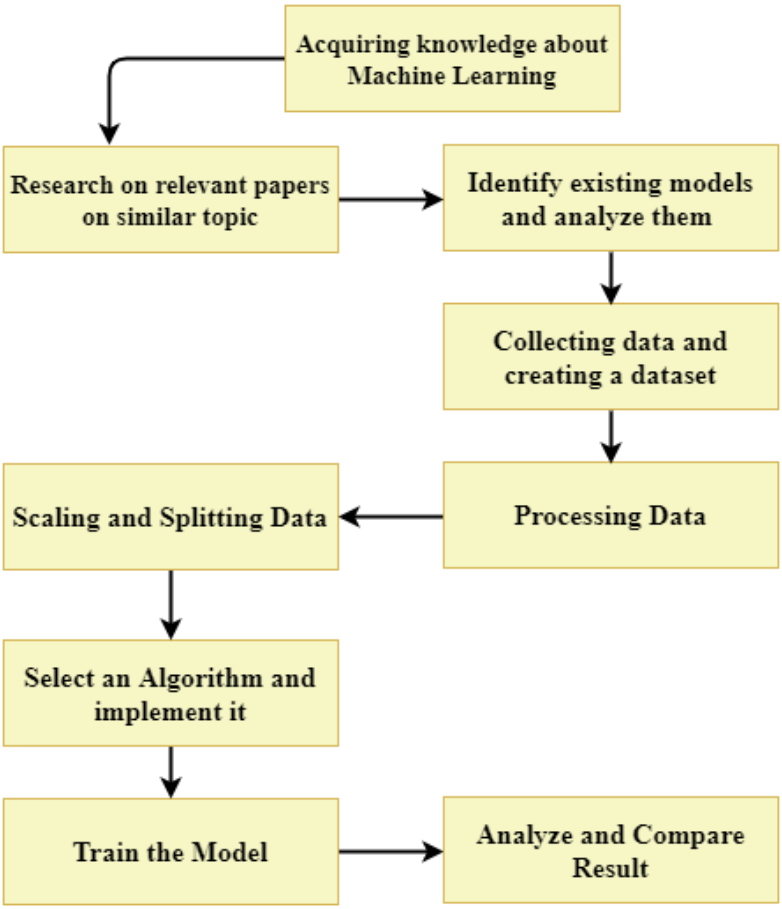


Figure 3.3: Workflow Diagram

Chapter 4

Data and Preliminary Analysis

In our dataset we have used data of different types of job advertisements for only Bangladesh which have been published from July to September of year 2020. We have collected all the data manually and for collecting data we have visited different kinds of websites like BD jobs, job.com.bd, bdjobstoday, deshijob etc. We have collected 4000 data where 3699 job recruitments are real and 301 job recruitments are fake. The percentage of fake jobs is 7.52%. For making our dataset we have used EMSCAD(Employment Scam Aegean Dataset) as reference.

4.1 Dataset Description

There are 2 different groups which are real jobs and fake jobs and from the fraudulent column we can find that. We have search for some selective parameters for every job and those are : Title, Location, Department, Salary range, Company profile, Job Description, Requirement, Benefits, Telecommunicating, Has Company Logo, Has Question, Employment type, Required experience, Required_education, Attractive words, fraudulent. We have collected different categories of job like : Accounting, finance, banking, education, engineering, garments, textile, HR, organizational, production, hospital, health, IT, telecommunication, marketing, media, NGO, research, security, law etc. so that our research would not be limited for some particular job.

In the title column we have entered the title of the job. In the location column where the office is located we collected the data. In the department column from which department they are offering are stored. The amount of salary is stored in the salary range column. Name of the company is stored in the Company profile. In Job Description we have mentioned the responsibility of that particular job holder. What are the requirements that they want from the applicants are mentioned in Requirement. The opportunities they are providing are stored in the benefits column. In the telecommunication column we have collected the information that there is any option of work from home. In the particular recruitment if there is any question ,is there any logo of that company, are they used any attractive words we have also collected those informations. What kind of educational and working experience

they want from the applicants we have also gathered those data. **Table 4.1** contains some sample data from our dataset.

Title	**Salary** range	**Company** profile	**Has** Company Logo	**Required** experience	**Attractive** words	**fraudulent**
Executive (Planning & Cordination)	Negotiable	Noman Group	0	2 to 4 year(s)	0	0
Shoe Last Modeler	Negotiable	hanghavi Shoe Accessories BD Pvt Ltd	1	At least 7 year(s)	0	0
Assistant Chemist	Negotiable	J. D. Golder & Co.	0	null	0	0
DGM/ Additional Chief Engineer (Electrical)	Negotiable	Deshbandhu Sugar Mills Ltd.	0	10 to 15 year(s)	0	0
Head of Factory/ Assistant General Manager (Factory Operation)	Negotiable	Navana Engineering Limited	0	At least 15 year(s)	0	0
CA Artcle/ Student	null	Islam Quazi Shafique & Co.	0	null	0	1

Table 4.1: Sample Dataset

In our dataset Telecommunicating, Has_Company_Logo, Has Question, Attractive words and fraudulent are integer data types and the rest of them are object data types. For the blank data in object data type we have replaced them with "null" and for integer data type we have replaced them with "0".

4.2 Data Pre-Processing

After collecting data our next step was to pre-process it for further analysis. In **Figure 4.1**, the dataset without any pre-processing is shown. In this step, the inaccurate, irrelevant and incomplete data will be removed and will be cleaned to an extent where the machine learning algorithm will understand the overall content of the data [18]. This step is very crucial for any kind of data analysis as it can have an effect on the accuracy of the result. As we are using both categorical data and numerical data we used different techniques to process the data.

	Title	Location	Department	Salary_range	Company_profile	Job Description	Requirement	Benefits
0	supervisor line chief	Gazipur	textile	NaN	elegant	keen knowledge operation breakdown machine layout	age 25 30years	nan
1	compilance office	Gazipur	textile	NaN	elegant	first one ever bangladesh produce color master...	postive attitude good communication skills	nan
2	cutting master cum trainer	Cox's Bazar	textile	30000	prottyashi	full responsibility quality function within lo...	least 05 years professional experience working...	nan
3	executive q c	Dhaka	textile	NaN	colorplus masterbatch manufacturing co ltd	first one ever bangladesh produce color master...	age 22 33yrs fresh graduates encouarged apply	2 festival bonuses extensive salary package

Figure 4.1: Data without Pre-processing

There were four steps in our pre-processing part. At first we separated the column that has categorical data. Then, we filtered out "stop words" from the data which is shown in **Figure 4.2**. In natural language processing "stop word" means the most common and useless words such as a, an, we, etc. These words can take unnecessary space and can take more processing time.

	Title	Location	Department	Salary_range	Company_profile	Job Description	Requirement	Benefits
0	Supervisor/Line Chief	Gazipur	Textile	NaN	Elegant	Keen knowledge about operation breakdown or ma...	Age 25-30years	NaN
1	Compilance Office	Gazipur	Textile	NaN	Elegant	We are the first one ever in Bangladesh to pro...	Postive attitude and good communication skills.	NaN
2	Cutting Master cum trainer	Cox's Bazar	Textile	30000	Prottyashi	Full responsibility for quality function withi...	At least 05 years of professional experience i...	NaN
3	Executive, Q.C	Dhaka	Textile	NaN	Colorplus Masterbatch Manufacturing Co. Ltd.	We are the first one ever in Bangladesh to pro...	Age 22-33yrs. Fresh graduates are encouarged t...	2 festival bonuses. Extensive salary package.

Figure 4.2: Data after removing stop word

After removing stop words from our data we processed our text data using another natural language processing named TF-IDF(Term Frequency-Inverse Document Frequency). TF-IDF is a very popular technique in NLP. It retrieves information from a document and state how important that particular word is for that document. At first, this technique transforms all the words into individual vectors. Then it retrieves features that work with vectors. The TF part calculates the frequency of a particular term against the total word count in total document and the IDF(Inverse Document Frequency) evaluates the weight of that term in the document. After that, with both TF and IDF values we count our final TF-IDF values for all the vectors. Then, we created a new column with extracted features from every column that has a text value.

Next, to pre-process the remaining data we used label encoder. It is a utility class that can convert data where only values between 0 and n_classes-1 will remain [19]. We used this technique for "Employment_type", "Location", "Salary_range" columns. First, We converted those object type data to string type data. Afterward, as shown in **Figure 4.3**, we transformed those string type data to numeric data. After that, we dropped the unnecessary column.

Location	Salary_range	Telecommunicating	Has_Company_Logo	Employment_type
17	220	0	0	1
17	220	0	0	1
11	148	0	1	0
13	220	0	1	1

Figure 4.3: Data after Label Encoding

The last step in our preprocessing was to scale features using standard scaler. Feature scaling is a technique where we can scale the independent features of our data in a fixed range. Standard scaler is a common and effective scaler. Standardizing data is also necessary for many machine learning estimators. Standard scaler rescales the features and transforms data in such a way where it has the mean value of 0 and standard deviation as 1. After standardizing our data we completed our pre-processing.

Chapter 5

Algorithms

Machine Learning Algorithms are used to predict the results. in this research we have used different algorithms to predict whether a job is fake or real. We experimented with some latest algorithms that have not been used frequently before to determine how they work on our dataset. Among the algorithms LightGBM, Gradient Boosting, Voting Classifier, AdaBoost have not been used as frequent as compared to other classifiers such as Random Forest, Decision Tree, Logistic Regression. The detailed description of the algorithm that we have used is given below:

5.1 Random Forest

In machine learning there are some well-known and useful classifiers, which have helped to strengthen these fields quite a lot. Such a classifier is Random Forest Tree. Random Forest is considered to be one of the most outstanding classifier algorithms. Through many studies, it has been proved that a random forest algorithm gives a high accuracy in prediction even with adequate amounts of abnormal values. RF was proposed by Breiman in 2001 which is a combination algorithm. Here if the result of the predicted value comes in discrete number, then we consider it to be random forest classification, whereas if the result comes in continuous value, then we say it is a random forest regression. A random forest f is a collection of T independent Decisions Trees, where $f = \mathrm{F}_1, \cdots, \mathrm{F}_\mathrm{t}, \cdots, \mathrm{F}_\mathrm{T}$. So, we should go through this algorithm in order to understand why it would be beneficial to use this algorithm, in order to get a right prediction.

The algorithm is divided into two stages. In the first stage it uses the bootstrap resampling method to create K training sets from the original dataset. Moreover, it keeps the size of each training set the same as the original dataset. After that for each of those bootstrap training sets, it creates a classification regression tree and k number of decision trees are produced and with those a forest is built. So, Decision Tree is another classifier which acts as the base of RF. So, Random Forest is made up of multiple Decision Trees and it takes a number of these Trees because it affects the decision-making conditions. This is necessary as different decision-

making conditions allow it to differentiate vague data properly. One such example is, when labels or titles are missing, it is hard to detect. So, this issue can be solved with multiple Decision Trees. Furthermore, with the growth of each tree, the branching process depends on random selection of m¡=M of all features, and so it does not select the best features as the node of the branches. So, it means that in the set of trees, all of them have equal probability of being sampled.

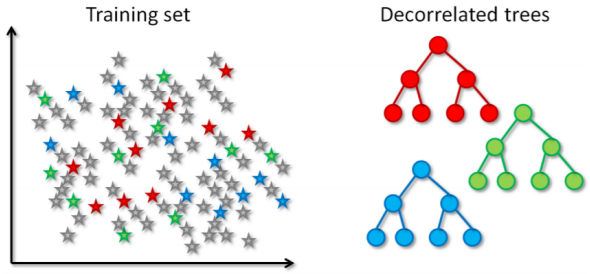


Figure 5.1: Bootstrap Aggregating [20]

Then in the second stage, the algorithm classifies the decision tree and a simple voting takes place. As all the Decision Trees are processed separately so the training process of Random Forest can take place concurrently which would boost the efficiency. To build the Random Forest, it combines K decision trees and trains them like before. Then the input samples are classified by seeing the result of a vote, that take place among the output of each Decision Tree. The algorithm finds the result by creating a number of independent and scattered Decision Trees and deciding the category of the final tree based on each Decision Tree. So, it can be said that Random Forest utilizes the multiple decision tree quite effectively as it picks those categories which dominate more, in different decision trees. Let's consider a random forest F, consisting of T Decision Trees, where $f = \{f_t\}_{t=1}^{T}$ Each of this tree generates a partition of the feature space χ, which is denoted as P_t. We consider individual tree as a surjective function with an an observation $\mathbf{X}\ \epsilon\ \chi$ to a cell ${C_t}^{(zt)}$of partition P_t , a function which is based on $\mathbf{X}$ to a group of cells for the whole forest is [20]:

$$f(X) = \left\{ {C_1}^{(zt)}, ..., {C_t}^{(zt)}, ..., {C_T}^{(zt)} \right\} \tag{5.1}$$

So the forest prediction can be determine by averaging the trees:

$$P(Y \mid X) = \frac{1}{T} \sum_{t=1}^{T} P(Y \mid X\ \epsilon\ {C_t}^{(zt)}, P_t) \tag{5.2}$$

Moreover, when it comes to process unrelated features data, Random Forest proves to be quite feasible. As our dataset contains different kinds of values and also it contains a number of unclear labels, so it was hard to predict the result. Hence, we have selected Random Forest as one of our classifiers which gives us an accuracy of

0.954. Also, it was troublesome to figure out the relation between the features. So, for all these reasons, Random Forest proves to be the appropriate classifier for our dataset, as it will process all the unrelated data and will provide us with adequate accuracy efficiently.

5.2 Voting Classifier

Voting classifier is one a popular algorithm of machine learning. It is quite a different kind of algorithm because it is actually not a particular algorithm, it is an ensemble machine learning algorithm. This classifier is considered as a meta model, which means model of models. It trains a number of different varieties of algorithms on it and those different algorithms predict different outputs and from all those outputs it chose the highest probability output. It basically combines all classifiers predictions and from the highest majority of voting it predicts the output. The main concept of this classifier is we do not need to find different separate classifiers, instead of that from only one classifier we can train different models and get the best prediction output based on the majority of voting from all the classifiers. But the voting system is not always the same. Either the ensemble involves the sum of the predictions which are made by other models or the average of all predictions. There are two types of voting system for this classifier:

1. Hard Voting:
   In this type of voting here the highest majority of votes got selected for the output. For example, If there are different types of classes like X Y Z and majority of classifiers predict output class X, then this hard voting will choose class X as output.

2. Soft Voting:
   In this type of voting the average result from all the classifiers will be selected as output. For example, if there are 3 classes X,Y,Z and each of them are given 2 models like X=(2,4), Y=(6,8), Z=(4,6). Therefore the averages for each of the classes will be, for X is 3, for Y is 7 and for Z is 5 and from these three classes the highest average is Y so this will be selected for output.

For using more models in this classifier the performance is better than any single model's performance. As it is a meta model so any kind of machine learning algorithm can be used on or with it and for this it does not need to import anything or need any kind of permission. For better performance this classifier needs more than one model. For the majority vote class it takes the output which is predicted by majority of the models. Here predictor space is X and rules for voting classifiers are h_1,h_2,h_3,....h_B [21].

$$C(X) = argmax_{\mathrm{i}} \sum_{j=1}^{B} \omega j I(h_{\mathrm{j}}(X) = i) \tag{5.3}$$

One disadvantage of this voting classifier is it treats all models equally for all situations but in reality all models are not perfect for every situation. Different models give the best solution in different situations.

5.3 LightGBM(Light Gradient Boosting) Classifier

LightGBM also known as LGBM is a gradient boosting framework originally developed by Microsoft for classification, ranking and other machine learning tasks. It is a free, open-source and very powerful algorithm for calculating and processing large amounts of data. The LightGBM classifier is based on decision tree algorithms and uses histogram based algorithms. This classifier also supports other algorithms like Gradient Boosted Regression Trees(GBRT), Random Forests, Dropouts meet Multiple Additive Regression Trees (DART), etc.

As LightGBM uses a histogram based algorithm, it has many advantages. One of them is less cost for calculation gain for each split [22]. The other pre sort based algorithm has the complexity of O(#data). The histogram based algorithm has the complexity of O(#data) and it reduces to O(#bin) after constructing the histogram which is much smaller than O(#data). The lightGBM can use histogram subtraction to speed up other leaf's histograms in a binary tree. It also reduces memory use and communication cost for parallel learning.

LightGBM uses Gradient-based One Side Sampling (GOSS) and Exclusive Feature Bundling (EFB) techniques to overcome the limitations of histogram based decision trees that most of the gradient boosting algorithms use [23]. These two techniques incorporate together to make the model more efficient. In the computation of information gain, different data instances have different roles. The GOSS technique takes the instances with larger gradients that will be more important for information gain and drops instances with smaller gradients randomly to maintain the accuracy of information gain estimation. For the same target sampling rate, this gives more accurate information gain than uniformly random sampling. The other EFB technique helps the model to reduce features by using a nearly lossless approach. A scattered feature space contains many mutually exclusive features. This technique safely bundles these exclusive features into a single feature. This changes the complexity of histogram building from O(#data $\times$ #feature) to O(#data $\times$ #bundle), while #bundle $\ll$ #feature and increases the training speed of the framework without affecting accuracy.

To optimize accuracy, LightGBM has a different approach for splitting its decision tree. Most other algorithms split the tree depth wise or level wise but LightGBM uses a leaf-wise approach to split the tree. It chooses the leaf which has maximum delta loss to grow [22]. So growing on a fixed leaf, leaf wise algorithm is more likely to have lower loss than the level-wise algorithms. As a result, it usually gives more accuracy than other existing boosting algorithms. The leaf wise approach may cause overfitting for smaller data. However, using the " max_depth" parameter can help

to solve this problem by limiting tree depth.

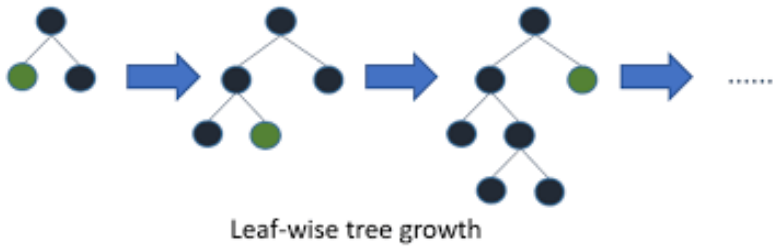


Figure 5.2: Light Gradient Boosting Machine [22]

To optimize the split of categorical data, LightGBM algorithm divides the features of categorical data into 2 subsets according to training objectives. If one category has K features there are $2^{(k-1)} - 1$ possible splits. So, LightGBM sorts the histogram of categorical data and calculates its accumulated value(sum_gradient / sum_hessian). Then it searches for the best split in the sorted histogram. LightGBM also optimizes network communication by using some collective communication algorithms.

5.4 Gradient Boosting

Gradient Boosting is one of the highly popular machine learning techniques that produces prediction models in the form of an ensemble of weak models. This algorithm trains many models in a gradual, additive and sequential manner. Through GBM(Gradient Boosting Machine) the mean squared error(MSE) is optimized [24]. When multiple weak models are combined to make a strong model, the MSE of the overall model is reduced.

The gradient boosting algorithm starts with initializing a model by first guess which is usually a decision tree which maximally reduces the MSE. Then at each step a new decision tree is added to the predictive model and fitted to the current residual to update the residual. The algorithm continues to iterate until the maximum number of iterations that is provided is reached. The shrinkage parameter is used to shrink the added decision tree at each step to provide better results. The shrinkage parameter takes values between 0 and 1 and choosing a smaller value implies stronger shrinkage and also implies that the number of iterations should be higher since the shrinkage parameter is inversely proportional to the number of iterations [25]. Gradient boosting can be described by the following equation [26]:

$$y = \mu + \sum_{m=1}^{M} \nu h_{\mathrm{m}}(y; X) + e; \tag{5.4}$$

where y is the vector of observed prediction, μ is the population mean, ν is the shrinkage parameter, $\mathrm{h_m}$ is the prediction model, X is the corresponding features of y and e is the residual.

5.5 AdaBoost

Boosting algorithms use the concept of combining simple classifiers. They turn the weak classifiers to stronger and by this they are boosting their performance. It has always been there but recently it is becoming more famous in problem solving cases for machine learning. One of the first boosting algorithms is Adaboost, which ensembles several weak classifiers and aims to convert into a stronger one. The full form of Adaboost is Adaptive Boosting, which was introduced by Freund and Schapire. It is so popular because it is extremely easy to use and gives quite productive results. The final equation for the classifier can be represented, as [27]:

$$F(x) = sign(\sum_{m=1}^{M} \theta_{\mathrm{m}} f_{\mathrm{m}}(x)) \tag{5.5}$$

Like weak classifiers we construct some "Decision Stumps" which would predict the same label for every example. Decision stumps are like trees in a forest but they have only 2 leaves and one root. Adaboost uses these stumps instead of a forest of trees. Depending on one feature the stump can be divided into 2 sets to improve the accuracy. They set a threshold and then place the examples on either side of the threshold. It can be tried with all the features and with different threshold values to see which decision stump fits the example properly and gives the most effective output. So, each decision stump gives a result on one input and gives an output of 1/-1 . From there an error is calculated and then it is modified to use the weighing of the training data. Then a stage value is generated for the training model. Along with it, for any prediction that the model makes a weighting is also provided. The weight of the weak classifier is generated by the given equation [27].

$$\theta_{\mathrm{m}} = \frac{1}{2} \ln(\frac{1 - \epsilon_{\mathrm{m}}}{\epsilon_{\mathrm{m}}}) \tag{5.6}$$

The optimizing process of Adaboost is greedy and it goes in a sequential manner. Only one weak classifier is added at a time. Suppose a training data is given, where weights are initialized for each data point. Then the dataset is fit onto the weak classifiers. If the accuracy is more than 50% then that classifier is considered to be more accurate and have more weight. On the other hand, a less accurate classifier has less weight. From there, the ones with the lowest weighted error are selected. And then calculate the weight of the weak classifier. It keeps on continuing until no improvements can be made or a pre-set number of weak learners is created. Once it is completed, we get a pool of weak learners with their stage values. The predictions that are made are based on the weighted average of the weak classifiers. As new inputs are given, each weak learner calculates a predicted value 1/-1 . These values are weighted with the stage value of the learners. The sum of the weighted prediction is given as the predicted value for the ensemble model. If the result of sum is negative then the second group is predicted but if it is positive then the first group is predicted [28]. In Adaboost, we get 1/-1 as the value of y, so the loss

function can be given as [27]:

$$E[e^{-yF(x)}] = e^{F(x)}P(y = -1 \mid x) + e^{-F(x)}P(y = 1 \mid x) \tag{5.7}$$

By solving it further we get,

$$\frac{\partial E[e^{-yF(x)}]}{\partial F(x)} = e^{F(x)}P(y = -1 \mid x) - e^{-F(x)}P(y = 1 \mid x) = 0, \tag{5.8}$$

$$F(x) = \frac{1}{2}\log(\frac{P(y = 1 \mid x)}{P(y = -1 \mid x)}) \tag{5.9}$$

The ensemble method tries to modify the miss classification in the data, so it's better to be careful that the data is good. As Adaboost helps to make the weak classifier stronger, so they are really flexible. Before it can be used for binary classifiers but it has been expanded to image and text classification as well.

5.6 Logistic Regression

Another powerful machine learning algorithm is logistic regression, it is quite popular in solving binary classification problems. The basic technique of the Logistic Regression algorithm is similar to linear regression. But the outcome of linear regression is continuous. It is based on probability concepts and uses a sigmoid function. The main idea of logistic regression is that it tends to keep the cost function between 0 or 1. Logistic variables should be used when the target variables are discrete and it should not be used with continuous target variables. A threshold is set at first, which determines what values should fall in one class vs other class. It calculates the relationship between the labels (dependent variable) and features (independent variable) by estimating probabilities. After that to make a prediction, these probabilities are converted into binary numbers. These values are then transformed to either 0 or 1, by using the threshold value and then placed to any one class. The image below portrays the stages logistic regression undergoes to give the desired result:

Also, it wants random data to be correctly classified, which is done by maximum likelihood estimation. It is a common way of estimating variables in statistical models. This technique is broadly used as it is quite efficient. As this algorithm can be implemented quite easily, it can be used as a baseline to estimate the performance for more complex algorithms. However, it fails to give a decision with non-linear problems. Another drawback of this algorithm is that its interpretation is harder, as the weights are multiplied. Function of logistic regression is [30]:

Figure 5.3: Logistic Regression [29]

$$logistic(\eta) = \frac{1}{1 + e^{(}-\eta)} \tag{5.10}$$

Below is the graph of logistic regression:

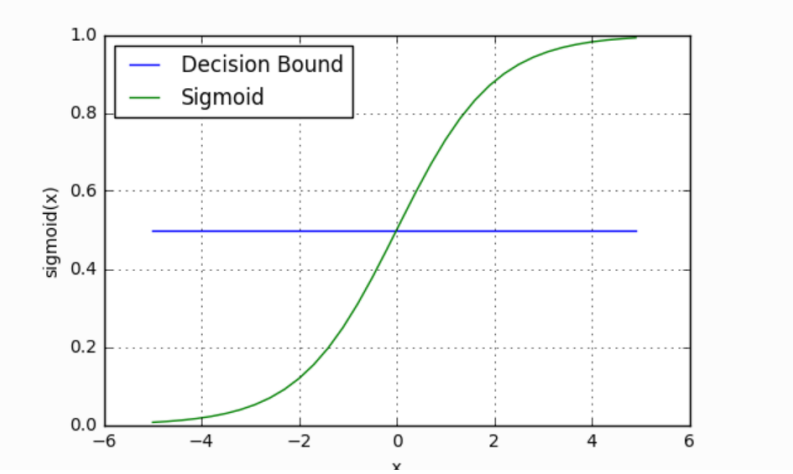


Figure 5.4: Logistic Regression Graph [31]

Furthermore, it can be used in multiclass classification problems and then it is called multinomial regression. To do this, it follows the "one vs all" approach. For instance, there are 3 groups in which the data can fall into, so it handles them separately and consider these groups as 3 different binary classifier problems.

5.7 Decision Tree Classifier

Decision tree is a path that goes like a tree structure and continues on the basis of state and rules. It has proven to be quite useful in the data science sector. This algorithm comes from the family of supervised learning algorithms. Regression and classification problems can also be solved with a decision tree. It is considered to be a flexible model because as we add more features on it, it does not affect the value of parameters. Also, the output can either be based on categorical or numerical prediction. So, in complex scenarios, it gives quite productive outcomes. The main

objective of the decision tree is that it can generate a training model based on learning decision rules, which helps to give a predicted class or a value of target variables. The tree contains two parts. One is a node and another is a branch. To make a prediction the data needs to go through a certain route, so for that each feature is assessed on each node. The decision tree is built recursively, assessing the features properly and then selecting the data for each node, that splits that data most accurately. To build the decision tree, a number of steps are required. Below shows a figure of the decision tree.

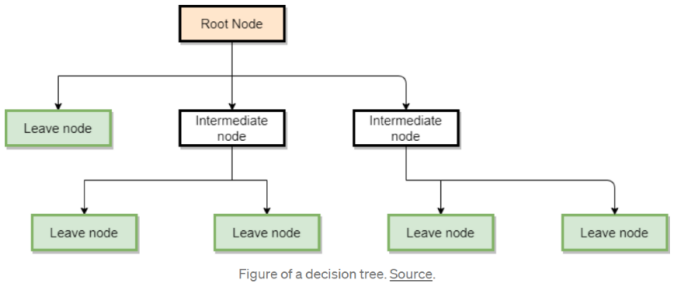


Figure 5.5: Decision Tree Classifier [32]

First comes splitting. This is the step where the dataset is divided into subsets. And these splits are generated on the basis of a particular variable. Then comes the second stage, which is called pruning. This stage contains the elimination of branches. By pruning, the size of the tree is reduced, as some branches are turned to leaf nodes. This technique is useful because the training data can be fit well by the trees. But for new values, it may not be that efficient. The next is tree selection. Through this process, the smallest tree that fits all the data is selected. Decision Tree is widely used because of its effectiveness and simplicity. Also, it can help in solving problems of a dataset that contains missing values. Furthermore, there is no need for a mathematical or statistical explanation of the outcome of decision trees. However, it is a bit difficult to interpret large trees.

Chapter 6

Result and Analysis

6.1 Feature Review

In our dataset there are 4000 jobs we have collected. Between those 4000 jobs 301 are fake and 3699 are real. Therefore, 92.47% jobs are real and 7.52% are fake jobs which is shown in the **Figure 6.1**.

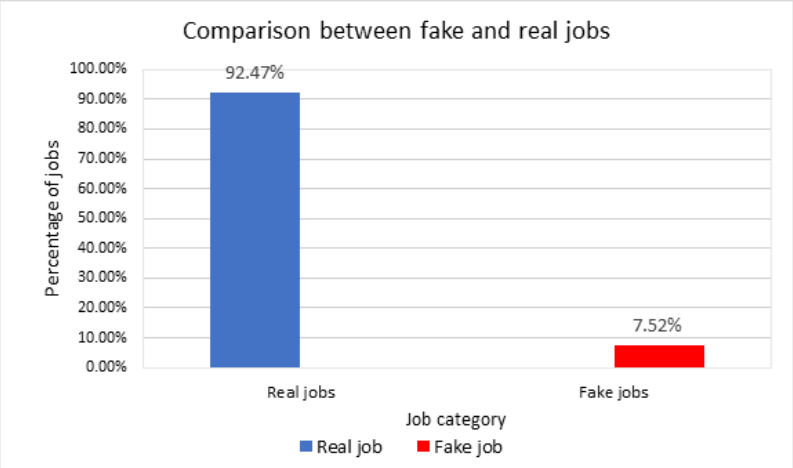


Figure 6.1: Amount of fake and real jobs

In our dataset there are 17 columns which means we have gathered 17 types of information from all job postings. But some of them are very important information because in fake jobs we have noticed that some of those information were missing or some information made them fake.

Firstly we can talk about the title and location of the jobs. We have noticed that all jobs always have a job title and almost every job they have mentioned about their job location but they have not given the address of the offices or companies or industries. They have only mentioned about a specific district or if the job is located in our capital city Dhaka sometimes they have mentioned about a specific area of Dhaka.

For having a good result we did not only collect jobs in a specific department. We wanted to analyse all kinds of jobs in Bangladesh. That is why we have collected jobs from various types of departments. Those are shown in **Figure 6.2**.

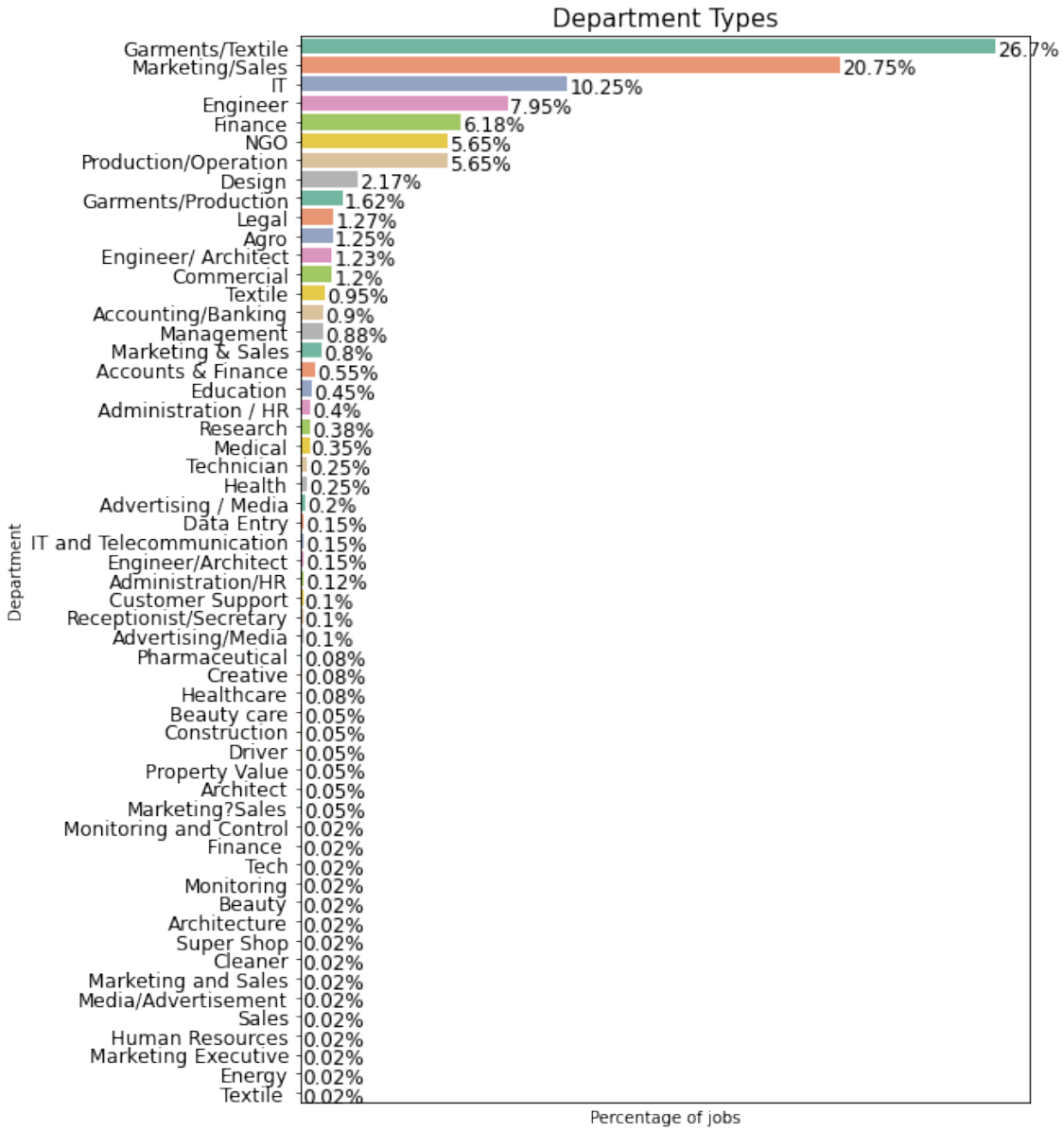


Figure 6.2: Types of departments and job numbers

In job postings we have found that in some postings they have mentioned something about jobs salary but in some other posts they have not mentioned anything about salary. Total 94.58% job posts, job salary or related salary was mentioned from there 88.58% were real jobs and 6.00% were fake, shown in **Figure 6.1**. On the other hand, 5.42% cases job salary was not there in posts and from them 1.52% was fake. Therefore we have noticed that if in a job post if there was not anything about salary the possibility of that job being fake is higher.

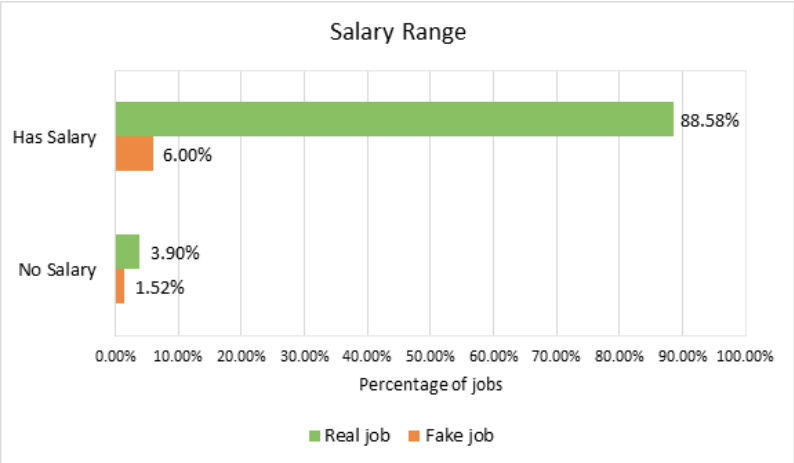


Figure 6.3: Salary range in fake and real jobs

For the company profile there were some jobs who have clearly mentioned the company name but in some jobs they have not given company names or have not clearly given the full name of the company. Here in **Figure 6.4**, a total of 97.9% job posts gave their company name where 6.1% jobs were fake. On other side the total number of jobs where company name was not mentioned was 2.1% and from there 1.42% are fake. And so we analyse that if a job does not have a company name then there is high probability that that job can be fake.

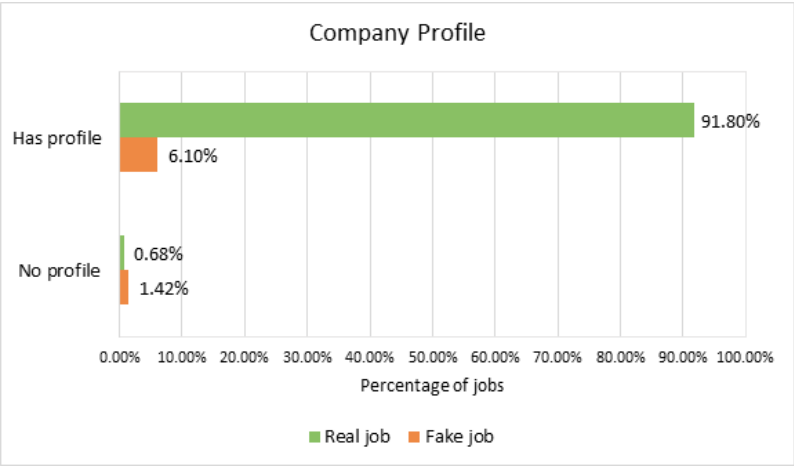


Figure 6.4: Company profile in fake and real jobs

In the job description part we have found in 96.38% jobs they have provided the responsibilities about that particular job and there 90.5% are real and only 5.88% are fake which is a very small amount, shown in **Figure 6.5**. Beside that 3.63% jobs have not provided any responsibilities about those jobs. Though it is a very small amount but from there 1.65% jobs are fake. For this reason we have found that if there is no job responsibility in a job post then the possibility of that job becoming fake is almost 50%.

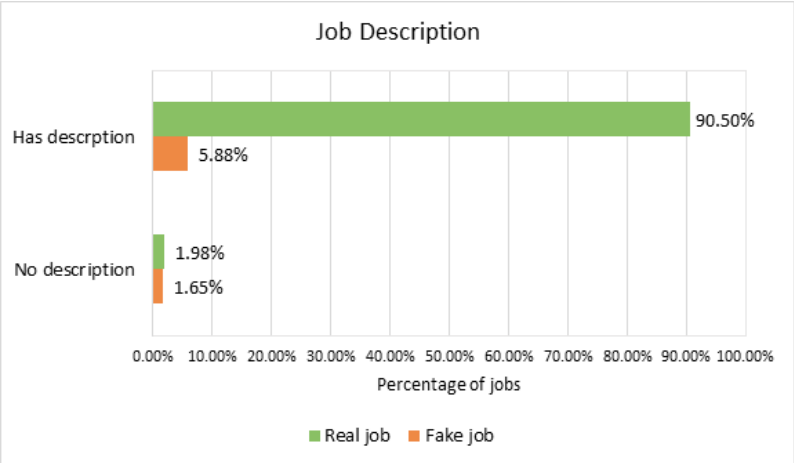


Figure 6.5: Job description in fake and real jobs

Apart from experience and educational requirements every employer wants some other requirement for that particular job and we collected those additional requirements in the requirements section. In **Figure 6.6**, we can see 96.25% job posts have given additional requirements where only a small amount is fake. At the same time, 3.74% of job posts have not mentioned additional requirements where almost half of the jobs are fake.

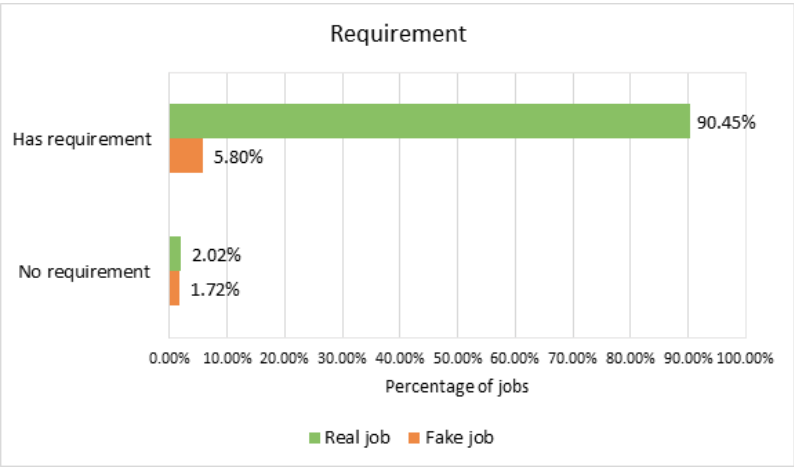


Figure 6.6: Requirements in fake and real jobs

Every candidate wants some benefits from their jobs. In **Figure 6.7**, it is seen that around 69.1% jobs have recommended benefits where the very minimum amount is fake 2.15%. But around 30.9% jobs have not recommended any benefit and there 5.38% are fake. Therefore in that view if a job is presenting benefits than it has very less possibility that it will be a fake job.

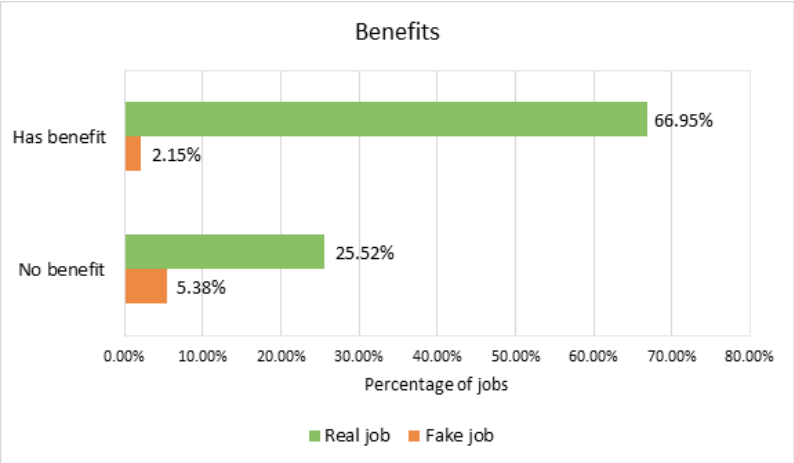


Figure 6.7: Benefits in fake and real jobs

Telecommuting is if a job gives an option to work from home. We have noticed that in Bangladesh jobs there are a very minimum amount of which give this work from home option which is 1.63% and from there only 0.25% jobs are fake which is not that much noticeable in **Figure 6.8**.

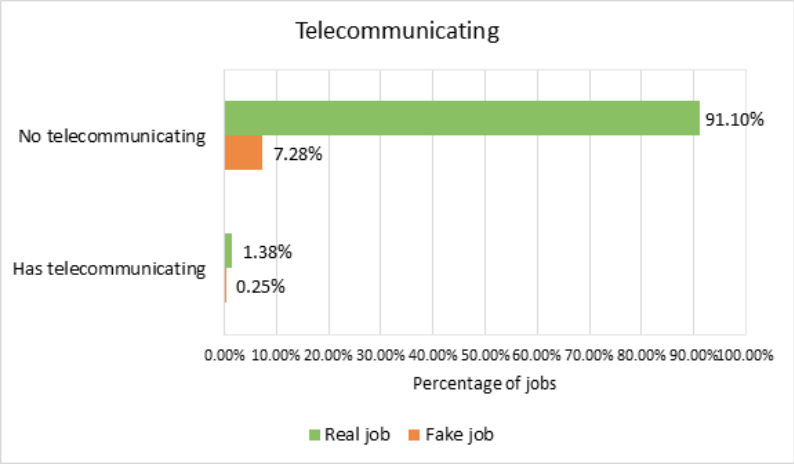


Figure 6.8: Telecommunicating in fake and real jobs

In Bangladeshi job posting we have noticed that very few jobs have given company logos in their job posting. In **Figure 6.9**, it is shown that around 17.75% job postings have company logos and on that only 0.48% jobs are fake. The amount of having no logo on job posting is very high which is 82.25% and there 7.05% jobs are fake.

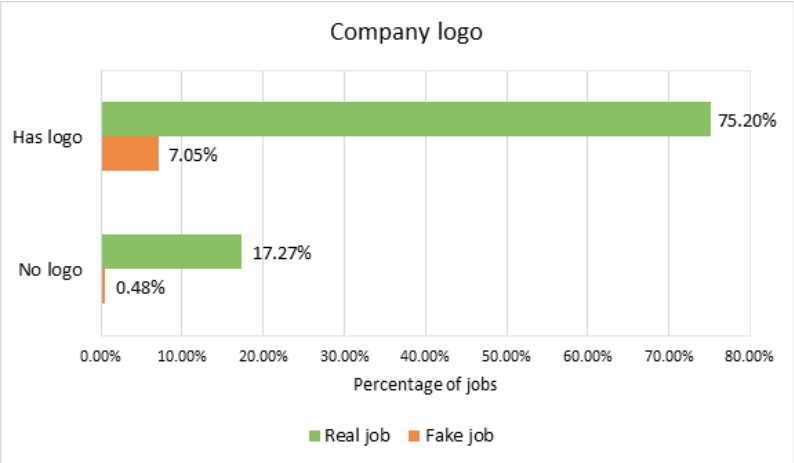


Figure 6.9: Company logo in fake and real jobs

One noticeable thing is in Bangladesh job posting there is no question. They haven't asked anywhere if a candidate has any question or not.

There are 5 common types of employment in job postings. Most visible is full time which is total 95.59% which is a very huge amount and from there 6.41% jobs are fake which is shown in **Figure 6.10**. Second type of employment is contractual which is 3.81%. Apart from these there are also part time, internship and other job types which are very minimal.

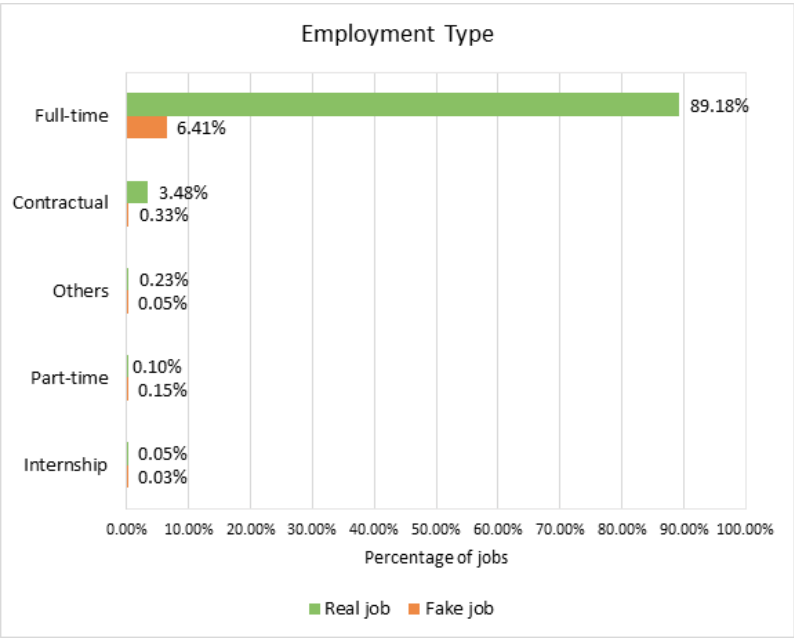


Figure 6.10: Different type of employment in fake and real jobs

For getting a good output from an employee, the employer wants some previous experience from that employee. Therefore, in **Figure 6.11** there are 90.05% of jobs where they have required experience and their 4.55% jobs are fake. On the other hand 9.95% of jobs have not required previous experience where 2.97% are fake which indicates if a job does not require work experience the chances of being fake

is quite high.

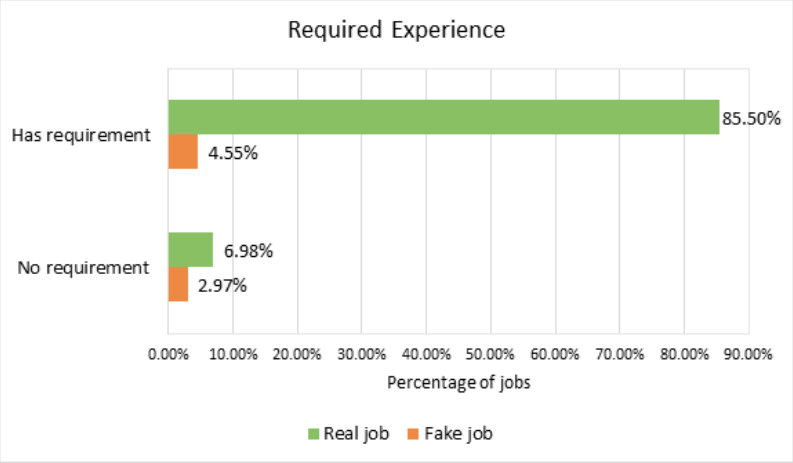


Figure 6.11: Required experience in fake and real jobs

It is mandatory that if a person wants to work which is not physical work and he/she must need to be educated. So a job posting education requirement is mandatory so that the employer will understand if the candidate is capable for the job or not. Most of the job postings require education. In **Figure 6.12** very little number of jobs around 1.33% jobs have not required education where 0.45% are fake jobs. But those who have required education, we have found some of the jobs require higher level education some wanted only SSC/HSC/A Levels/O levels which are 4.2% where 1% is fake. Therefore, we analyse if a job requires little amount of education then there is one fourth possibility that that job can turn into a fake job. Anyway, there are 94.48% where they have required higher level education which means most of the time they required higher degrees.

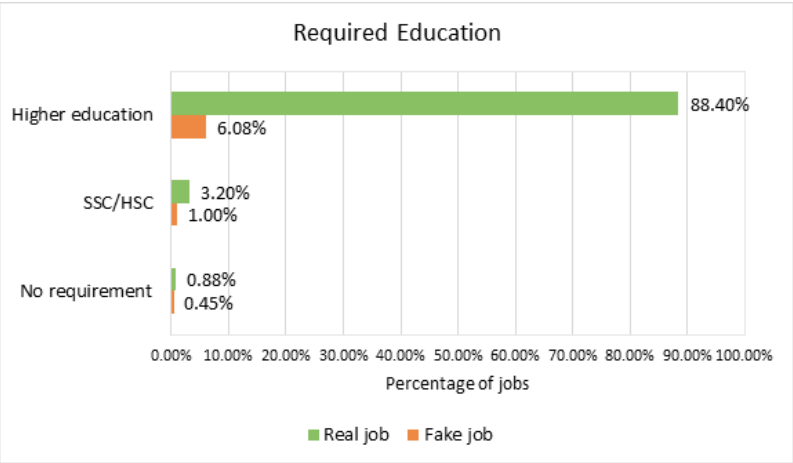


Figure 6.12: Required education in fake and real jobs

Sometimes to get attention from a job seeker in job posting they provide some attractive words. But in Bangladeshi online job posting most of the cases which is 96.72% there were not any attractive words which is shown in **Figure 6.13**. On the other side only 3.27% job postings have provided attractive words where 0.62% are fake.

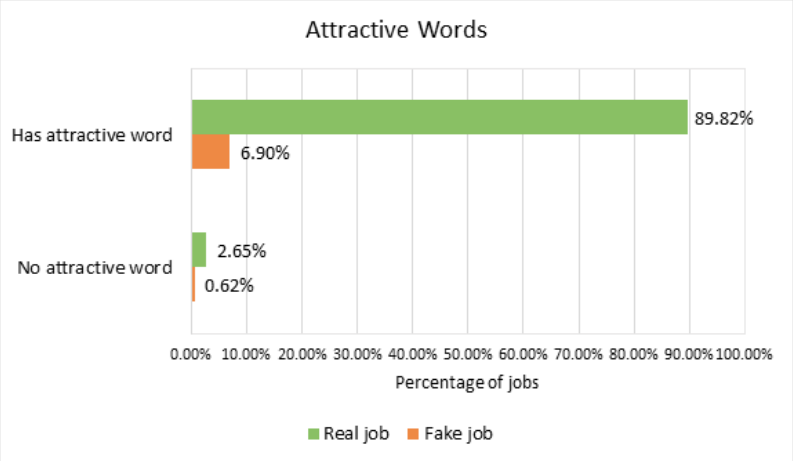


Figure 6.13: Attractive words in fake and real jobs

In data processing we have gathered all strings together of all the jobs and we have observed that in real jobs text lengths are greater and in fake jobs they use small amounts of text in job postings which is shown in **Figure 6.14**.

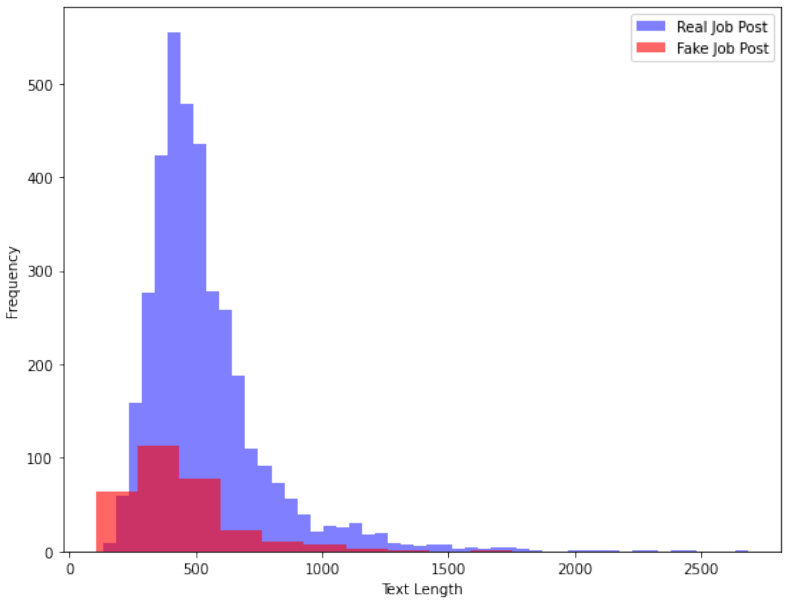


Figure 6.14: Text Length in Real and Fake jobs

In a nutshell our analysis says that if in a job posting they do not mention about salary or job responsibilities or benefits , if there is no company name or logo, they do not require work or education or additional experience, if allow to works from home or they use attractive words in their posting then the probability of being that job fake is higher.

6.2 Data Visualization

Data Visulaization is an easy going way to understand the patterns in data through charts, tables, graphs or maps etc. It is a graphical representation of the data. In python many libraries like matplotlib, seaborn, plotly, ggplot are some of the many notable libraries that are widely used to plot heatmap, pie chart, scatter and density plot, bar graph etc.

1. **H**eatmap
   Over here in **Figure 6.15**, to plot the heatmap we have used the python seaborn library. Heatmap is basically a colored matrix representation where each block of color is dependent on the data. The blocks with higher value get a hot color, whereas the block with less value gets a cold color. Here the block for fraudulent and Required_experience is showing a hot region so it means that their value is high.

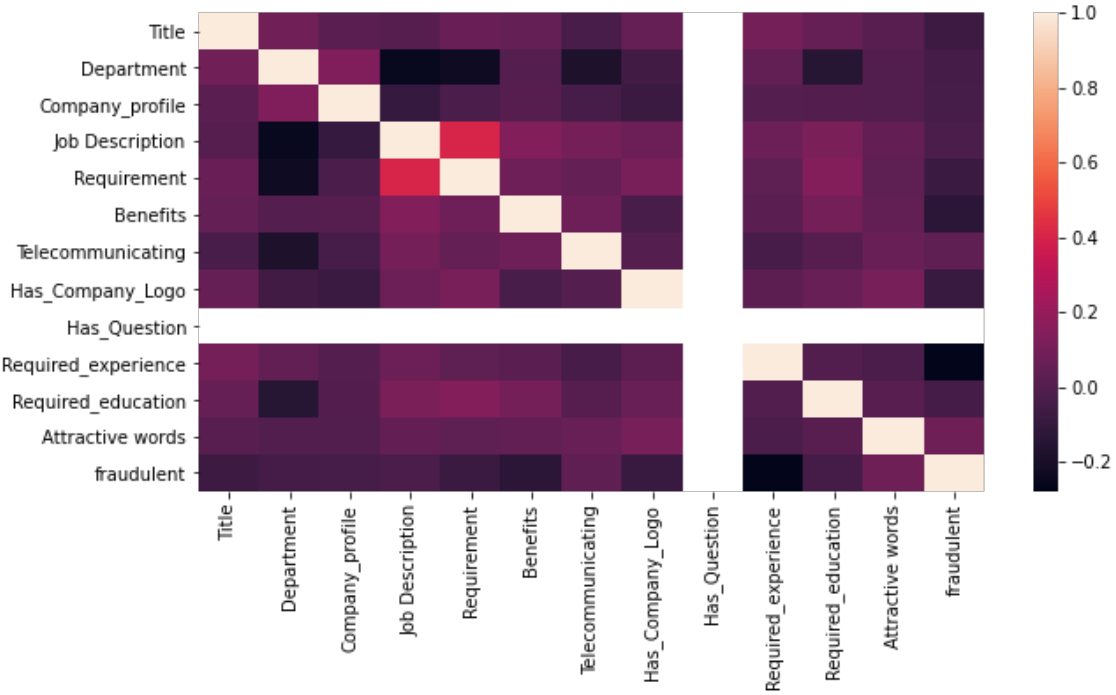


Figure 6.15: Heatmap

2. **P**ie Chart
One of the most suitable ways of data visualization is through pie charts. In the pie chart, each slice of circle shows a portion of the data. And it is quite easy to understand statistical data by seeing the chart.

When we were collecting our data to create our dataset we noticed that not all information is provided in all job postings. In many job postings different kinds of information were not mentioned.
When we found that the information we were looking for was not mentioned then for string type of information we considered those as null values. In **Figure 6.16** here are the percentages of null values for string type data. For string data we found more null values in "benefit" and "required experience". In every job posting other string dates like "title", "location", "department" were mentioned, we did not find any null value for these kinds of data.

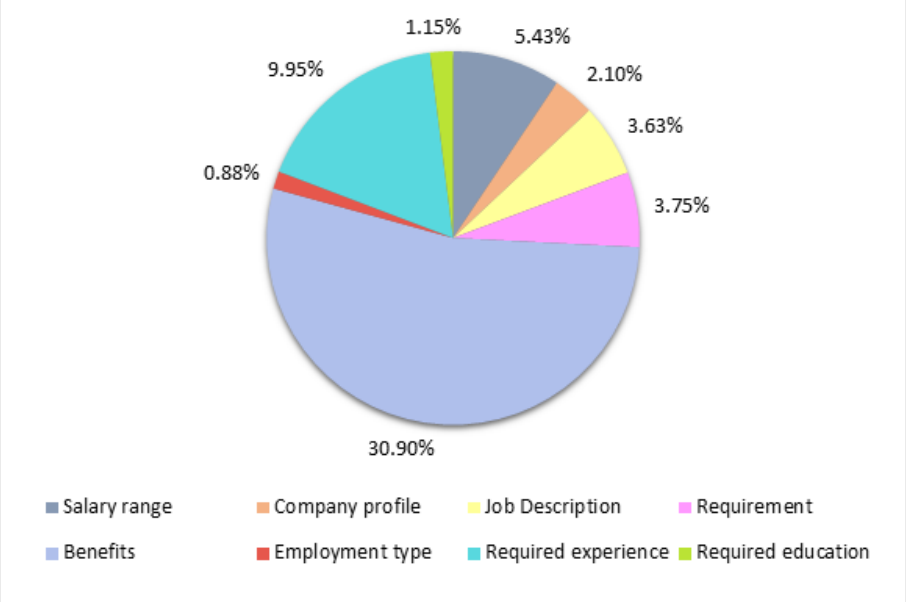


Figure 6.16: Percentages of null values for string type data

For the data we were collecting as binary form there for missing information we considered that as 0. In **Figure 6.17** here are the percentages of 0 values for binary type data. For binary most of the cases it was 0 as most of the time we did not find information in job posting. We also have another binary type data which is "has question" but we in our dataset all values for "has question" is 0 as we did not find any question in our job posting. Therefore we did not count that data type.

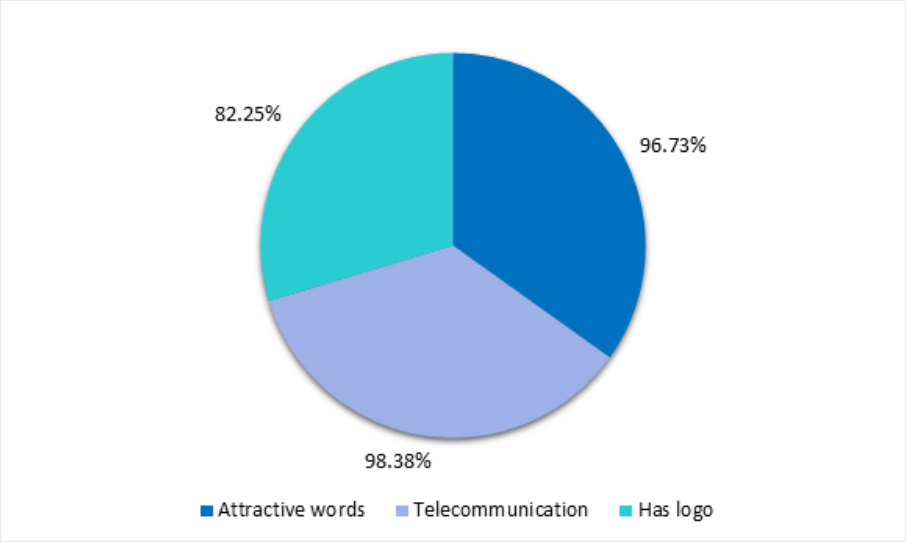


Figure 6.17: Percentages of 0 values for binary type data

In **Figure 6.18**, we have done a pie chart to represent each percentage of employment type. So, from the diagram we can see that 95.59% of the job is full-time. And also, other jobs percentages are shown accordingly.

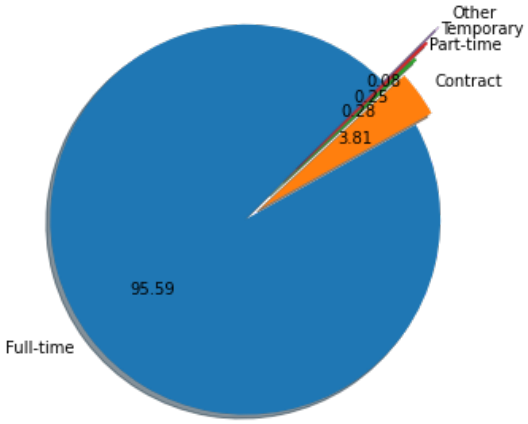


Figure 6.18: Percentage of employment type

3. **A**xes Plot
   When we want to use multiple axes in one figure then we should use an axes plot. To generate this, we should use subplot. An axes object is returned from axes.plot(). All the axes objects has a parent figure. So, in the first plot, we have shown the comparison between department and company profile with respect to fraud in **Figure 6.19**. Over here, the blue line represents the department and orange line represent the company profile.

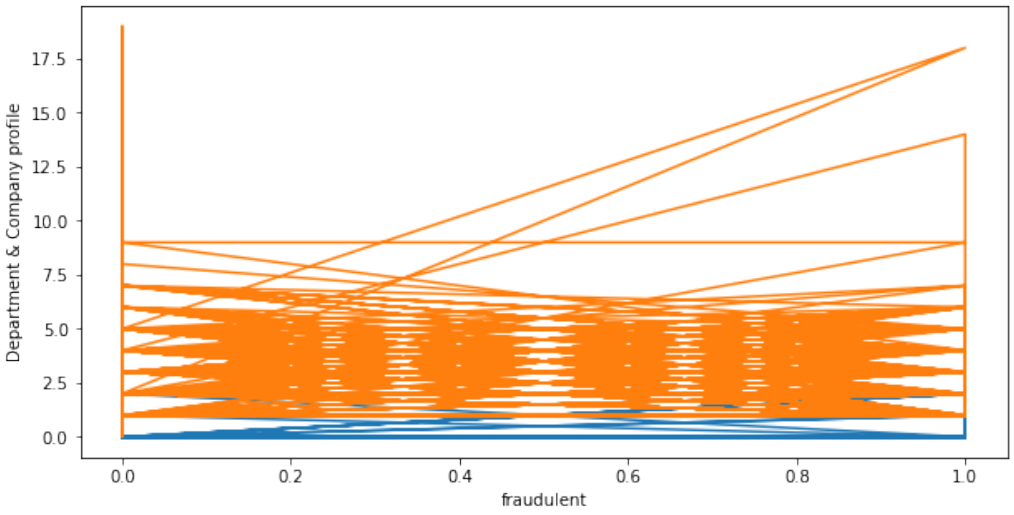


Figure 6.19: Comparison between department and company profile with respect to fraud

Then in the second plot we used the axes.plot() to represent three features in one figure. Over here the requirement is blue line, required education is orange and required experience is shown by green. And all these are compared with respect to fraudulent. From the **Figure 6.20** we can see that the green line has less fraudulent value. So, we can say that required experience has less effect in fraud compared to other too.

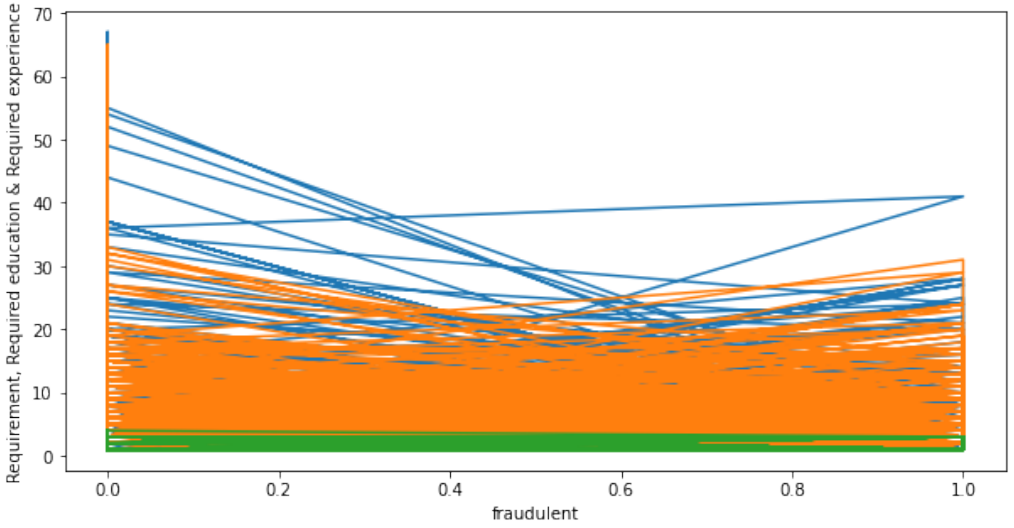


Figure 6.20: Comparison between requirement, required education and required experience with respect to fraud

Then in the third diagram **Figure 6.21**, we have shown a comparison between job description and benefits. These are done with respect to fraudulent. Then the blue line represents the job description and the orange line shows the benefit. So, from here we can say that job description is more related to fraudulent.

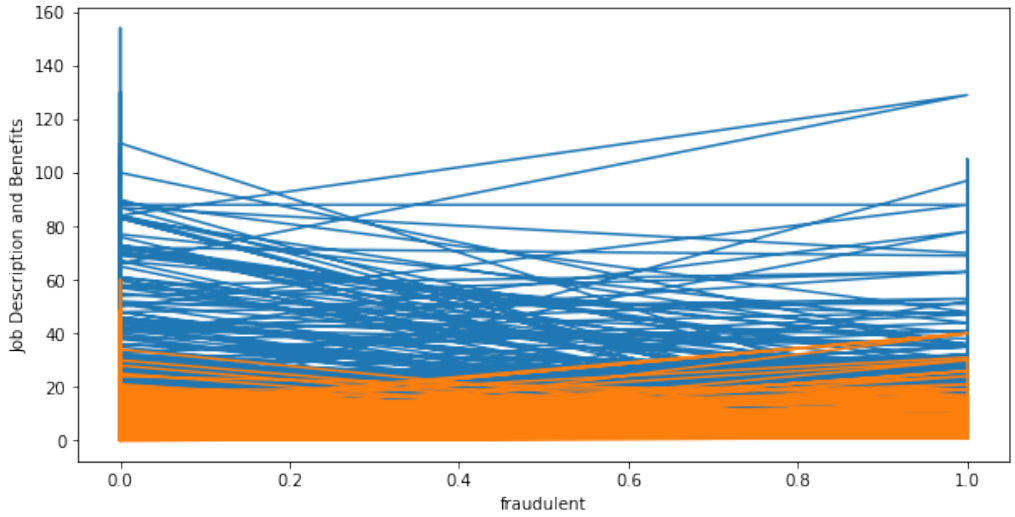


Figure 6.21: Comparison between job description and benefits with respect to fraud

6.3 Algorithm Accuracy Comparison

We used seven algorithms to detect which provides better result on our dataset. Those algorithms are: Random forest Classifier, Voting Classifier, LightGBM Classifier, Gradient Boosting, AdaBoost, Logistic Regression and Decision Tree. We used both common and latest classifiers to determine the difference in the accuracy level of those algorithms. From the results we have seen that there is not much difference in the accuracy of those algorithms. The accuracy range is between 94% to 96%. We also experimented on the dataset that we used as a reference to see how well does that behave on the aforementioned algorithms. It is to be noticed in **Table 6.1** that this dataset has more accuracy on the algorithms and they range from 97% to 99%. The accuracy comparison table of different algorithm on our dataset and the referenced dataset is given below:

Algorithm Name	**Result on our dataset**	**Result on referenced dataset**
Random Forest	95.00%	98.92%
Voting Classifier	95.34%	98.24%
LightGBM	95.17%	99.22%
AdaBoost	95.00%	97.99%
Gradient Boosting	95.17%	99.25%
Logistic Regression	94.67%	96.72%
Decision Tree Classifier	95.00%	97.99%

Table 6.1: Accuracy Comparison Table

From the above table, it is clearly noticed that the dataset that has been used as a reference works better on the experimented algorithms. But even though we have used the EMSCAD dataset as reference, we have found some dissimilarities between both the datasets. We have analyzed that those differences are the cause of different accuracy results as well. The major differences that we have found in the datasets

are listed below:

- The first thing we have figured is that there are three columns in the EMSCAD dataset that have no relevant information in our dataset. The columns industry and function have not been used in our dataset as there is no relevant information posted in the job advertisements in Bangladesh. Thus, these two columns had been removed from our dataset. Furthermore, the EMSCAD dataset has a column named job_id which has also not been used in our dataset. The accuracy of different algorithms when these columns have been removed from referenced dataset is given below in **Table 6.2**:

Algorithm Name	**Result on referenced dataset**
Random Forest	97.24%
Voting Classifier	97.04%
LightGBM	98.37%
AdaBoost	96.00%
Gradient Boosting	97.04%
Logistic Regression	96.12%
Decision Tree Classifier	96.00%

Table 6.2: Accuracy Table after removing industry, function and job_id from reference dataset

- We have used the column attractive words which take string as an input. This column takes in words that have been used in the job advertisements to attract the job seekers towards applying to their company. Words like attractive salary, attractive remuneration, attractive salary package, faster career growth are used to attract the job seekers. To check whether this column creates an effect on the accuracy level of our model, we removed it and passed the modified dataset to check the accuracy which is shown in **Table 6.3**:

Algorithm Name	**Result on our dataset**
Random Forest	95.00%
Voting Classifier	95.55%
LightGBM	94.98%
AdaBoost	95.37%
Gradient Boosting	95.16%
Logistic Regression	95.75%
Decision Tree Classifier	95.37%

Table 6.3: Accuracy Table after removing attractive words from our dataset

  We can see that removing the column results in further decrease in the accuracy of the algorithms.

- It is also seen that most of the companies whether fake or real do not have any company logo given in the job advertisements. From what has been mentioned

before, 82.25% job advertisements do not post any logo along with other details. That is why it is very difficult to discriminate between real and fake jobs using this feature. For this reason we removed company logo from both datasets to determine the change in accuracy levels which is shown in **Table 6.4**:

Algorithm Name	**Result on our dataset**	**Result on referenced dataset**
Random Forest	95.00%	97.08%
Voting Classifier	95.34%	97.97%
LightGBM	94.86%	98.39%
AdaBoost	95.00%	96.62%
Gradient Boosting	94.83%	97.23%
Logistic Regression	95.14%	96.72%
Decision Tree Classifier	95.00%	96.62%

Table 6.4: Accuracy Comparison Table after removing company_logo

We can see that the removal of company logo feature affected the accuracy level of the reference dataset by a relatively higher margin than it affected our created dataset.

- In the EMSCAD datasheet there is a column named Company profile where the details of the company have been provided. In our dataset no such details about the company has been provided except the name. That is why the company name column has been used in our dataset instead of company profile. After removing company profile feature from the EMSCAD dataset, the updated accuracy of the algorithms is given in **Table 6.5**:

Algorithm Name	**Result on referenced dataset**
Random Forest	97.65%
Voting Classifier	97.04%
LightGBM	97.41%
AdaBoost	96.28%
Gradient Boosting	97.04%
Logistic Regression	96.12%
Decision Tree Classifier	96.15%

Table 6.5: Accuracy Table after removing company profile from reference dataset

- In the dataset that we have created most of the job posts have mentioned salary range as negotiable. Very few jobs have provided either a specific range of amount or that section has been empty. But the reference dataset, the advertisements that have mentioned salary range has been filled and the rest others are empty which means that the data in the salary range column of EMSCAD dataset is either empty or with a salary range. Whereas the salary range column in our dataset has either negotiable, null or a specific range of value. The accuracy of algorithms when salary range has been removed from both the datasets is given in **Table 6.6**:

Algorithm Name	Result on our dataset	Result on referenced dataset
Random Forest	95.00%	97.42%
Voting Classifier	95.34%	97.43%
LightGBM	95.17%	98.53%
AdaBoost	95.00%	96.61%
Gradient Boosting	94.84%	97.86%
Logistic Regression	95.37%	96.29%
Decision Tree Classifier	95.00%	96.61%

Table 6.6: Accuracy Table after removing salary range

We can see that the removal of salary range feature from our dataset has not created much impact on the accuracy level of the algorithms but it has created a relatively greater impact when it was removed from the reference dataset.

- Lastly, the major difference between both the datasets is the size. The number of data in the EMSCAD dataset is 17,881 while the number of data in our dataset is 4000. Size of the dataset can also create major difference in the accuracy of algorithms. That is why we have selected 4000 data from the EMSCAD dataset randomly to determine the accuracy level of different algorithms.Before applying algorithms we have removed the column company_logo from both our dataset and referenced dataset since in our dataset this column is mostly empty but in the referenced dataset this column is comparatively less empty. Here we have found that for 4000 random dataset there is a huge change in the referenced dataset's accuracy. For all algorithms the accuracy level has decreased. Even the decreasing level is this much that our dataset's accuracy is better than their dataset's accuracy for almost all algorithms. In **Table 6.7** the comparison between two datasets when company logo has been removed is given below where the number of data in the referenced dataset is 4000:

Algorithm Name	Result on our dataset	Result on referenced dataset
Random Forest	95.00%	94.16%
Voting Classifier	95.34%	94.33%
LightGBM	94.86%	95.83%
AdaBoost	95.00%	93.67%
Gradient Boosting	94.83%	94.52%
Logistic Regression	95.14%	94.00%
Decision Tree Classifier	95.00%	93.67%

Table 6.7: Accuracy Comparison Table after removing company_logo where size of referenced dataset is 4000

All these above mentioned reasons have created a huge impact on the overall dataset that we have created which is why we can see dissimilarities in the accuracy result of both the datasets as well.

Chapter 7

Conclusion and Future Works

In our research we have compared the performance of different machine learning algorithms like Random Forest, Voting Classifier, LGBM, Gradient Boosting, Logistic Regression, AdaBoost, Decision Tree Classifier and determined which algorithm works best on our dataset. We did not only use the most common algorithms but also some latest ones so that we can determine how well do they work. We have seen that range of accuracy of all our algorithms lie between 94% to 96%. The main purpose for our model is to help the job seekers identify which jobs are fake and save themselves from fraudsters. Our model can also be used by different online job recruitment sites to detect fraudulent jobs. We have identified that some features play an important role in determining whether a job is fake or not. In future, we plan to incorporate these features in our dataset and thus try to further increase the accuracy range of the algorithms. Also, we would try to add data from other job sites in Bangladesh as well so that we could cover almost all job posting in Bangladesh. In this way we would not only have a bigger dataset but also achieve better algorithm performance.

Bibliography

[1] V. Stilwell, *There are now more than five million job openings in america*, 2015. [Online]. Available: https://www.bloomberg.com/news/articles/2015-02-10/job-openings-in-u-s-rose-by-181-000-in-december-to-5-03-million.

[2] J. Howington. (2015). "Survey: More millennials than seniors victims of job scams," [Online]. Available: https://www.flexjobs.com/blog/post/survey-results-millennials-seniors-victims-job-scams/.

[3] Z. A. Palak, *Digital bangladesh - a story of transformation*, 2019. [Online]. Available: https://www.daily-sun.com/post/407497/Digital-Bangladesh--a-Story-of-Transformation.

[4] 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. 1, p. 6, 2017.

[5] S. Morgan and M. Park, "Cybercrime report 2017," 2017. [Online]. Available: https://www.herjavecgroup.com/cybercrime-report-2017/.

[6] "Personal fraud," 2014-2015. [Online]. Available: https://www.abs.gov.au/statistics/people/crime-and-justice/personal-fraud/latest-release.

[7] S. Lal, R. Jiaswal, N. Sardana, A. Verma, A. Kaur, and R. Mourya, "Orfdetector: Ensemble learning based online recruitment fraud detection," in *2019 Twelfth International Conference on Contemporary Computing (IC3)*, 2019, pp. 1–5.

[8] B. Alghamdi and F. Al-Harby, "An intelligent model for online recruitment fraud detection," *Journal of Information Security*, vol. 10, pp. 155–176, 2019.

[9] T. Suryanto and R. R. A. Saleh, "Employee recruitment fraud prevention with the implementation of decision support system," *Journal of Physics: Conference Series*, vol. 1028, p. 012 055, Jun. 2018.

[10] S. Vidros, C. Kolias, and G. Kambourakis, "Online recruitment services: Another playground for fraudsters," *Computer Fraud Security*, vol. 2016, pp. 8–13, Mar. 2016.

[11] S. Mahbub and E. Pardede, "Using contextual features for online recruitment fraud detection," Aug. 2018.

[12] S. G. Thiago and M. C. Walmir, "A review of machine learning approaches to spam filtering," *Expert Systems with Applications*, vol. 36, no. 7, pp. 10 206–10 222, 2009, ISSN: 0957-4174.

[13] J. R. Scanlon and M. S. Gerber, "Automatic detection of cyber-recruitment by violent extremists," *Security Informatics*, vol. 3, no. 1, p. 5, 2014.

[14] K. Shivagangadhar, H. Sagar, S. Sohan, and C. Vanipriya, "Fraud detection in online reviews using machine learning techniques," *International Journal of Computational Engineering Research (IJCER)*, vol. 5, no. 5, pp. 52–56, 2015.

[15] D. Sculley and G. M. Wachman, "Relaxed online svms for spam filtering," in *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '07, Amsterdam, The Netherlands: Association for Computing Machinery, 2007, pp. 415–422.

[16] Newtium, *Steps of predictive modelling.* [Online]. Available: https://newtiummedia.blob.core.windows.net/images/Steps-to-Predictive-Modelling.jpg (visited on 10/04/2020).

[17] I. S. Candanedo, E. H. Nieves, S. R. González, M. T. S. Martın, and A. G. Briones, *Machine learning predictive model for industry 4.0*, Springer, 2018.

[18] A. Kafi, M. S. A. Alam, S. B. Hossain, S. B. Awal, and H. Arif, "Feature-based mobile phone rating using sentiment analysis and machine learning approaches," in *2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT)*, IEEE, 2019, pp. 1–6.

[19] L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. VanderPlas, A. Joly, B. Holt, and G. Varoquaux, "API design for machine learning software: Experiences from the scikit-learn project," in *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, 2013, pp. 108–122.

[20] O. Pauly, "Random forests for medical applications," Ph.D. dissertation, Technische Universität München, 2012.

[21] G. James, "Majority vote classifiers: Theory and applications," 1998.

[22] *Welcome to lightgbm's documentation!* 2018. [Online]. Available: https://lightgbm.readthedocs.io/en/latest/Features.html#references.

[23] *Lightgbm (light gradient boosting machine)*, 2020. [Online]. Available: https://www.geeksforgeeks.org/lightgbm-light-gradient-boosting-machine/ (visited on 12/25/2020).

[24] T. Parr and J. Howard, *How to explain gradient boosting.* [Online]. Available: https://explained.ai/gradient-boosting/index.html (visited on 12/25/2020).

[25] S. Touzani, J. Granderson, and S. Fernandes, "Gradient boosting machine for modeling the energy consumption of commercial buildings," *Energy and Buildings*, vol. 158, pp. 1533–1543, 2018.

[26] O. González-Recio, J. Jiménez-Montero, and R. Alenda, "The gradient boosting algorithm and random boosting for genome-assisted evaluation in large data sets," *Journal of dairy science*, vol. 96, no. 1, pp. 614–624, 2013.

[27] *Boosting algorithm: Adaboost*, 2017. [Online]. Available: https://towardsdatascience.com/boosting-algorithm-adaboost-b6737a9ee60c?gi=f8043aeb37ba (visited on 12/25/2020).

[28] K. Vidhya and R. Shanmugalakshmi, "Improved diabetic data analytic model for complication prediction," *Int. J. Eng. Adv. Technol*, 2019.

[29] N. Donges, *The logistic regression algorithm*, 2019. [Online]. Available: https://www.experfy.com/blog/bigdata-cloud/the-logistic-regression-algorithm/ (visited on 12/25/2020).

[30] *Logistic regression*, 2019. [Online]. Available: https://ml-cheatsheet.readthedocs.io/en/latest/logistic_regression.html (visited on 12/25/2020).

[31] *Logistic regression*. [Online]. Available: https://christophm.github.io/interpretable-ml-book/logistic.html (visited on 12/25/2020).

[32] J. Zornoza, *Decision trees explained*, 2020. [Online]. Available: https://towardsdatascience.com/decision-trees-explained-3ec41632ceb6 (visited on 12/25/2020).