1. INTRODUCTION

Employment scam is one of the serious issues in recent times addressed in the domain of Online Recruitment Frauds (ORF). In recent days, many companies prefer to post their vacancies online so that these can be accessed easily and timely by the job-seekers. However, this intention may be one type of scam by the fraud people because they offer employment to job-seekers in terms of taking money from them. Fraudulent job advertisements can be posted against a reputed company for violating their credibility. These fraudulent job post detection draws good attention for obtaining an automated tool for identifying fake jobs and reporting them to people for avoiding application for such jobs. For this purpose, a machine learning approach is applied which employs several classification algorithms for recognizing fake posts. In this case, a classification tool isolates fake job posts from a larger set of job advertisements and alerts the user. To address the problem of identifying scams on job posting, supervised learning algorithms as classification techniques are considered initially. A classifier maps input variables to target classes by considering training data. Classifiers addressed in the paper for identifying fake job posts from the others are described briefly. These classifier based predictions may be broadly categorized into -Single Classifier based Prediction and Ensemble Classifiers based Prediction.

1.1 Objectives

The main objective is to detect the fake job post, which is a classic text classification problem with a straightforward proposition. It is needed to build a model that can differentiate between a “Real” job post and “Fake” job post.

1.2 Methodology

To predict job posts a large collection of the company’s job posts are required. Employment Scam Aegean Dataset (EMSCAD) dataset is used to predict fake job posts. In this section the methodology followed is discussed in detail**.**

1.2.1 Dataset

Dataset collection :

Data is a set of records. This step is concerned with selecting the subset of all available data. EMSCAD dataset is used to train ML algorithms. Employment Scam

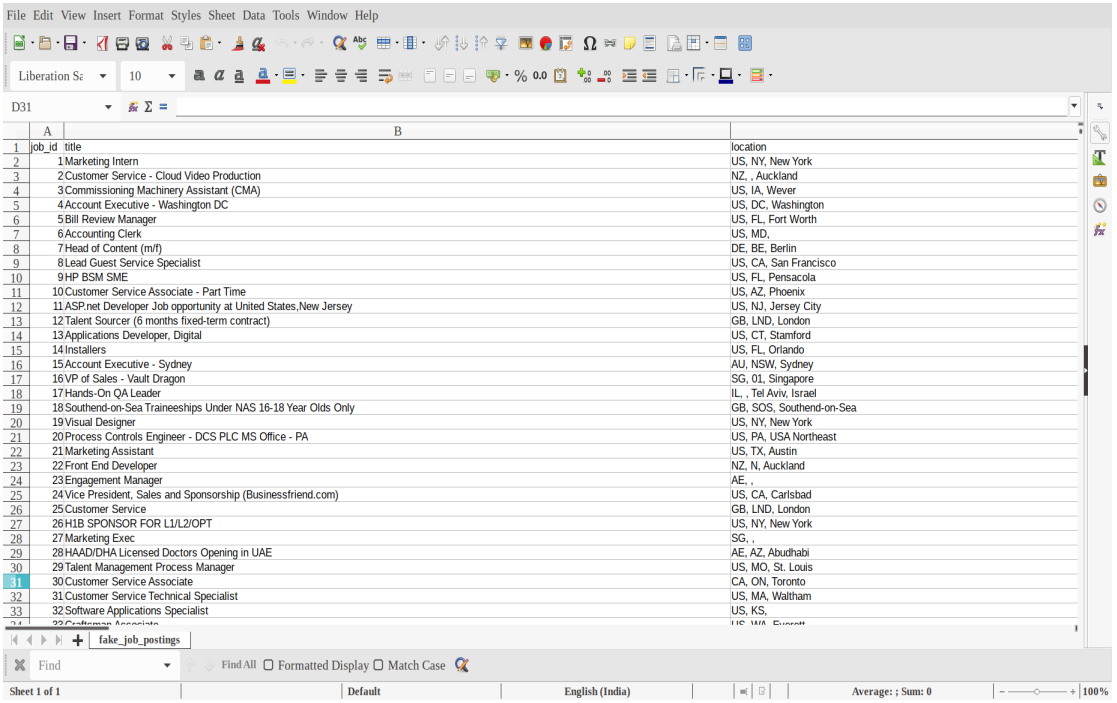
Aegean Dataset (EMSCAD) dataset which is provided publicly by the University of the Aegean Laboratory of Information & Communication Systems Security. This dataset contains 17,880 real-life job postings in which 17,014 are real and 866 are fake. Dataset contains labelled data.

Attribute	Description
Job_id	Unique Job ID
Title	The title of the job ad entry
Location	Geographical location of the job ad
Department	Corporate department
Salary_range	Indicative salary range
Company_profile	A brief company description.
Description	The details description of the job ad.
Requirements	Enlisted requirements for the job opening
benefits	Enlisted offered benefits by the employer.

Fig 1.2.1.1: Dataset - Attributes with description

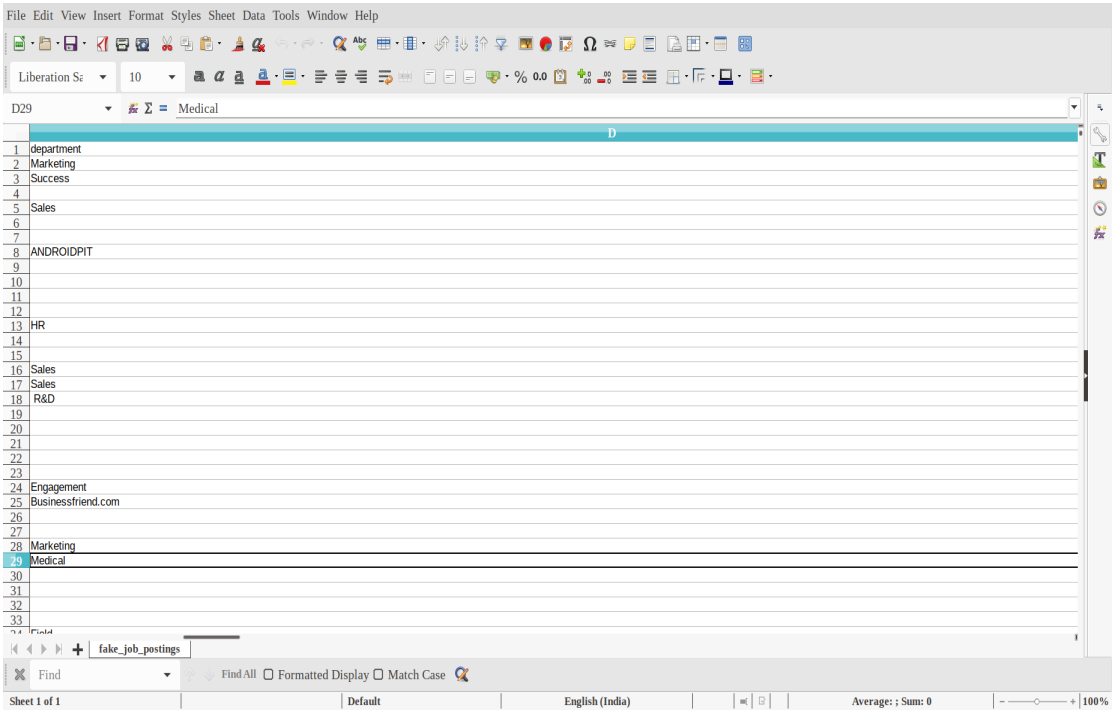
Attribute	Description
telecommuting	True for telecommuting positions.
has_company_logo	True if company logo is present.
has_questions	True if screening questions are present.
employment_type	Full-type, Part-time, Contract, etc.
required_experience	Executive, Entry level, Intern, etc.
required_education	Doctorate, Master's Degree, Bachelor, etc
industry	Automotive, IT, Health care, Real estate, etc.
function	Consulting, Engineering, Research, Sales etc.
fraudulent	target - Classification attribute.

Fig 1.2.1.2: Dataset - Attributes with description



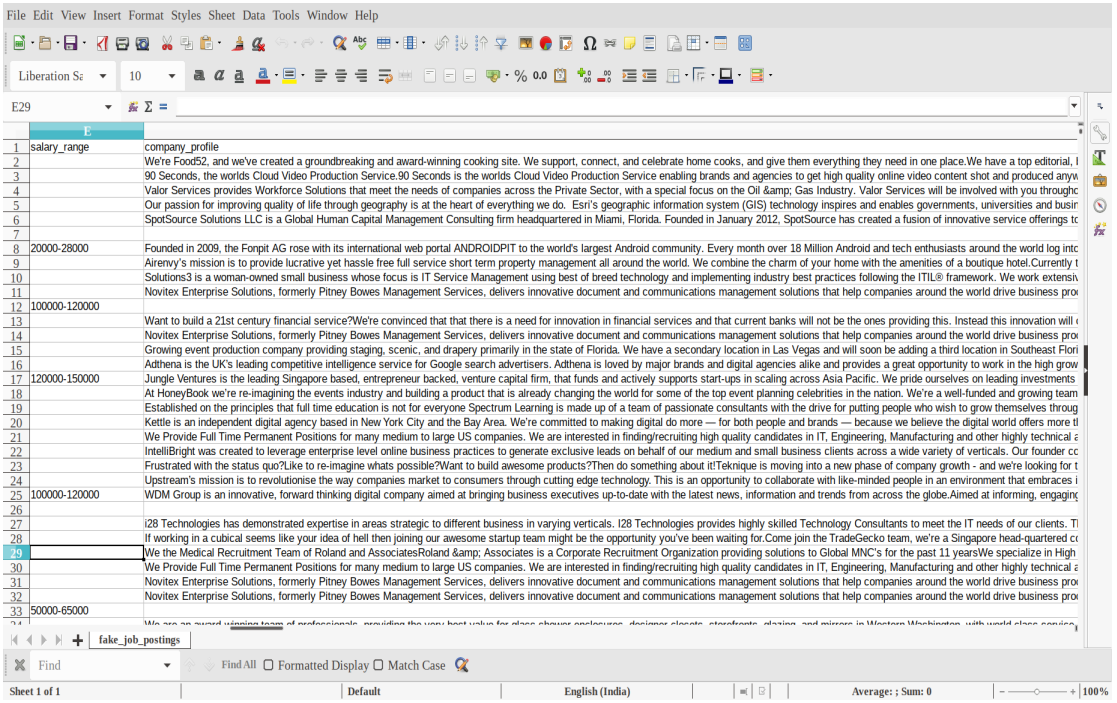
	A	B	
1	job_id	title	location
2	1	Marketing Intern	US, NY, New York
3	2	Customer Service - Cloud Video Production	NZ, , Auckland
4	3	Commissioning Machinery Assistant (CMA)	US, IA, Wever
5	4	Account Executive - Washington DC	US, DC, Washington
6	5	Bill Review Manager	US, FL, Fort Worth
7	6	Accounting Clerk	US, MD,
8	7	Head of Content (m/f)	DE, BE, Berlin
9	8	Lead Guest Service Specialist	US, CA, San Francisco
10	9	HP BSM SME	US, FL, Pensacola
11	10	Customer Service Associate - Part Time	US, AZ, Phoenix
12	11	ASP.net Developer Job opportunity at United States,New Jersey	US, NJ, Jersey City
13	12	Talent Sourcer (6 months fixed-term contract)	GB, LND, London
14	13	Applications Developer, Digital	US, CT, Stamford
15	14	Installers	US, FL, Orlando
16	15	Account Executive - Sydney	AU, NSW, Sydney
17	16	VP of Sales - Vault Dragon	SG, 01, Singapore
18	17	Hands-On QA Leader	IL, , Tel Aviv, Israel
19	18	Southend-on-Sea Traineeships Under NAS 16-18 Year Olds Only	GB, SOS, Southend-on-Sea
20	19	Visual Designer	US, NY, New York
21	20	Process Controls Engineer - DCS PLC MS Office - PA	US, PA, USA Northeast
22	21	Marketing Assistant	US, TX, Austin
23	22	Front End Developer	NZ, N, Auckland
24	23	Engagement Manager	AE, ,
25	24	Vice President, Sales and Sponsorship (Businessfriend.com)	US, CA, Carlsbad
26	25	Customer Service	GB, LND, London
27	26	H1B SPONSOR FOR L1/L2/OPT	US, NY, New York
28	27	Marketing Exec	SG, ,
29	28	HAAD/DHA Licensed Doctors Opening in UAE	AE, AZ, Abudhabi
30	29	Talent Management Process Manager	US, MO, St. Louis
31	30	Customer Service Associate	CA, ON, Toronto
32	31	Customer Service Technical Specialist	US, MA, Waltham
33	32	Software Applications Specialist	US, KS,

Fig 1.2.1.3: Sample dataset of job_id, title, location attributes



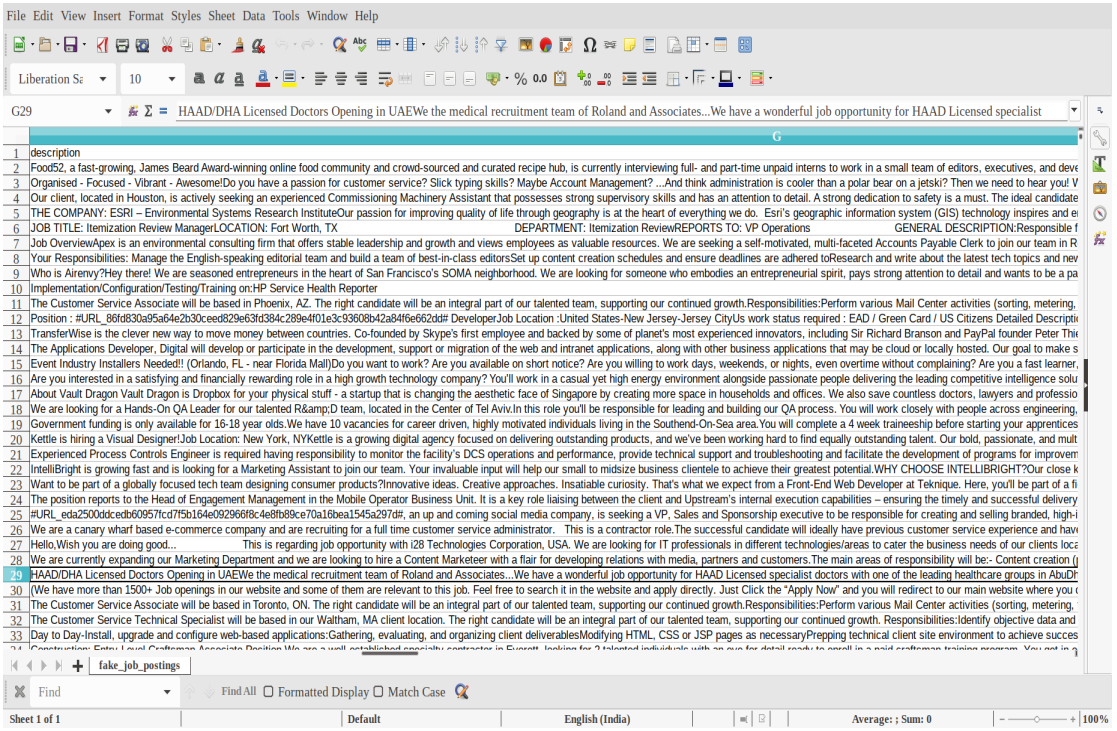
	D
1	department
2	Marketing
3	Success
4	
5	Sales
6	
7	
8	ANDROIDPIT
9	
10	
11	
12	
13	HR
14	
15	
16	Sales
17	Sales
18	R&D
19	
20	
21	
22	
23	
24	Engagement
25	Businessfriend.com
26	
27	
28	Marketing
29	Medical
30	
31	
32	
33	

Fig 1.2.1.4: Sample dataset of department attribute



	E
1	salary_range
2	
3	
4	
5	
6	
7	
8	20000-28000
9	
10	
11	
12	100000-120000
13	
14	
15	
16	
17	120000-150000
18	
19	
20	
21	
22	
23	
24	
25	100000-120000
26	
27	
28	
29	
30	
31	
32	
33	50000-65000

Fig 1.2.1.5: Sample dataset of salary_range and company_profile attributes



	G
1	description
2	Food52, a fast-growing, James Beard Award-winning online food community and crowd-sourced and curated recipe hub, is currently interviewing full- and part-time unpaid interns to work in a small team of editors, executives, and deve
3	Organised - Focused - Vibrant - Awesome!Do you have a passion for customer service? Slick typing skills? Maybe Account Management? ...And think administration is cooler than a polar bear on a jetski? Then we need to hear you! V
4	Our client, located in Houston, is actively seeking an experienced Commissioning Machinery Assistant that possesses strong supervisory skills and has an attention to detail. A strong dedication to safety is a must. The ideal candidate
5	THE COMPANY: ESRI – Environmental Systems Research InstituteOur passion for improving quality of life through geography is at the heart of everything we do. Esri's geographic information system (GIS) technology inspires and e
6	JOB TITLE: Itemization Review ManagerLOCATION: Fort Worth, TX DEPARTMENT: Itemization ReviewREPORTS TO: VP Operations GENERAL DESCRIPTION:Responsible f
7	Job OverviewApex is an environmental consulting firm that offers stable leadership and growth and views employees as valuable resources. We are seeking a self-motivated, multi-faceted Accounts Payable Clerk to join our team in R
8	Your Responsibilities: Manage the English-speaking editorial team and build a team of best-in-class editorsSet up content creation schedules and ensure deadlines are adhered toResearch and write about the latest tech topics and nev
9	Who is Airenvy?Hey there! We are seasoned entrepreneurs in the heart of San Francisco's SOMA neighborhood. We are looking for someone who embodies an entrepreneurial spirit, pays strong attention to detail and wants to be a pa
10	Implementation/Configuration/Testing/Training on:HP Service Health Reporter
11	The Customer Service Associate will be based in Phoenix, AZ. The right candidate will be an integral part of our talented team, supporting our continued growth.Responsibilities:Perform various Mail Center activities (sorting, metering,
12	Position : #URL_86fd830a95a64e2b30ceed829e63fd384c289e4f01e3c93608b42a84f6e662dd# DeveloperJob Location :United States-New Jersey-Jersey CityUs work status required : EAD / Green Card / US Citizens Detailed Descripti
13	TransferWise is the clever new way to move money between countries. Co-founded by Skype's first employee and backed by some of planet's most experienced innovators, including Sir Richard Branson and PayPal founder Peter Thi
14	The Applications Developer, Digital will develop or participate in the development, support or migration of the web and intranet applications, along with other business applications that may be cloud or locally hosted. Our goal to make s
15	Event Industry Installers Needed!! (Orlando, FL - near Florida Mall)Do you want to work? Are you available on short notice? Are you willing to work days, weekends, or nights, even overtime without complaining? Are you a fast learner,
16	Are you interested in a satisfying and financially rewarding role in a high growth technology company? You'll work in a casual yet high energy environment alongside passionate people delivering the leading competitive intelligence solu
17	About Vault Dragon Vault Dragon is Dropbox for your physical stuff - a startup that is changing the aesthetic face of Singapore by creating more space in households and offices. We also save countless doctors, lawyers and professio
18	We are looking for a Hands-On QA Leader for our talented R&D team, located in the Center of Tel Aviv.In this role you'll be responsible for leading and building our QA process. You will work closely with people across engineering,
19	Government funding is only available for 16-18 year olds.We have 10 vacancies for career driven, highly motivated individuals living in the Southend-On-Sea area.You will complete a 4 week traineeship before starting your apprentices
20	Kettle is hiring a Visual Designer!Job Location: New York, NYKettle is a growing digital agency focused on delivering outstanding products, and we've been working hard to find equally outstanding talent. Our bold, passionate, and mult
21	Experienced Process Controls Engineer is required having responsibility to monitor the facility's DCS operations and performance, provide technical support and troubleshooting and facilitate the development of programs for improvem
22	IntelliBright is growing fast and is looking for a Marketing Assistant to join our team. Your invaluable input will help our small to midsize business clientele to achieve their greatest potential.WHY CHOOSE INTELLIBRIGHT?Our close k
23	Want to be part of a globally focused tech team designing consumer products?Innovative ideas. Creative approaches. Insatiable curiosity. That's what we expect from a Front-End Web Developer at Teknique. Here, you'll be part of a fi
24	The position reports to the Head of Engagement Management in the Mobile Operator Business Unit. It is a key role liaising between the client and Upstream's internal execution capabilities – ensuring the timely and successful delivery
25	#URL_eda2500ddcedb60957fcd7f5b164e092966f8c4e8fb89ce70a16bea1545a297d#, an up and coming social media company, is seeking a VP, Sales and Sponsorship executive to be responsible for creating and selling branded, high-i
26	We are a canary wharf based e-commerce company and are recruiting for a full time customer service administrator. This is a contractor role.The successful candidate will ideally have previous customer service experience and have
27	Hello,Wish you are doing good... This is regarding job opportunity with i28 Technologies Corporation, USA. We are looking for IT professionals in different technologies/areas to cater the business needs of our clients loca
28	We are currently expanding our Marketing Department and we are looking to hire a Content Marketeer with a flair for developing relations with media, partners and customers.The main areas of responsibility will be:- Content creation (
29	HAAD/DHA Licensed Doctors Opening in UAEWe the medical recruitment team of Roland and Associates...We have a wonderful job opportunity for HAAD Licensed specialist doctors with one of the leading healthcare groups in AbuDh
30	(We have more than 1500+ Job openings in our website and some of them are relevant to this job. Feel free to search it in the website and apply directly. Just Click the "Apply Now" and you will redirect to our main website where you c
31	The Customer Service Associate will be based in Toronto, ON. The right candidate will be an integral part of our talented team, supporting our continued growth.Responsibilities:Perform various Mail Center activities (sorting, metering,
32	The Customer Service Technical Specialist will be based in our Waltham, MA client location. The right candidate will be an integral part of our talented team, supporting our continued growth. Responsibilities:Identify objective data and
33	Day to Day-Install, upgrade and configure web-based applications:Gathering, evaluating, and organizing client deliverablesModifying HTML, CSS or JSP pages as necessaryPrepping technical client site environment to achieve succes

Fig 1.2.1.6: Sample dataset of description attribute

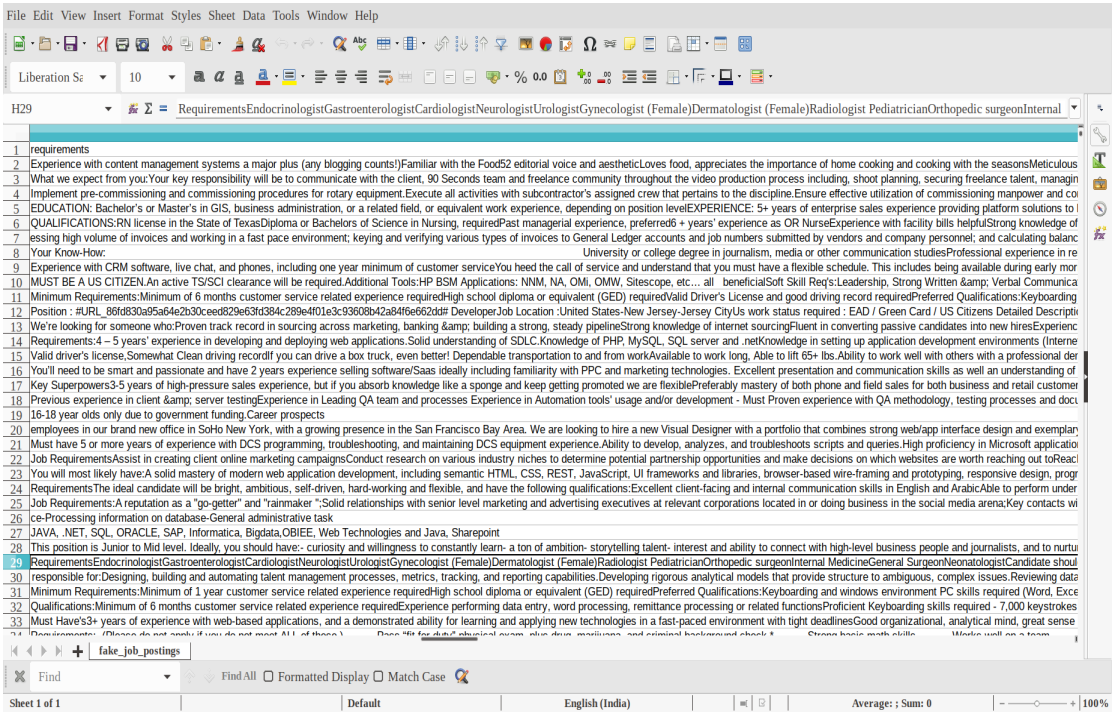


Fig 1.2.1.7: Sample dataset of requirements attribute

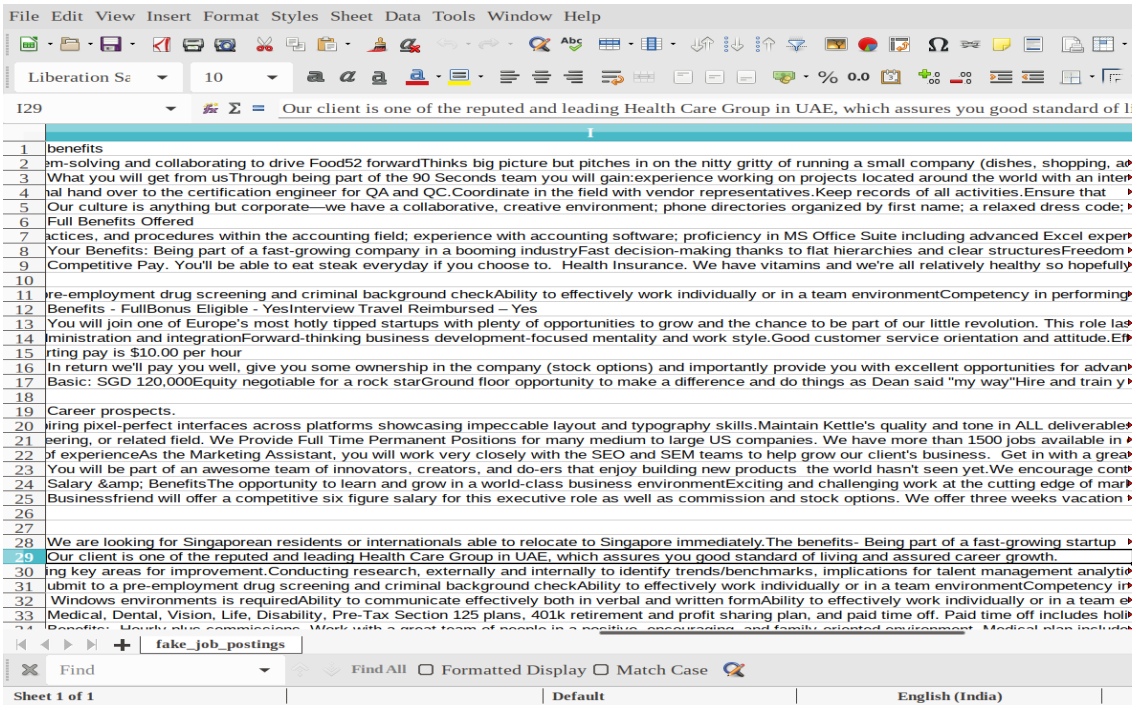
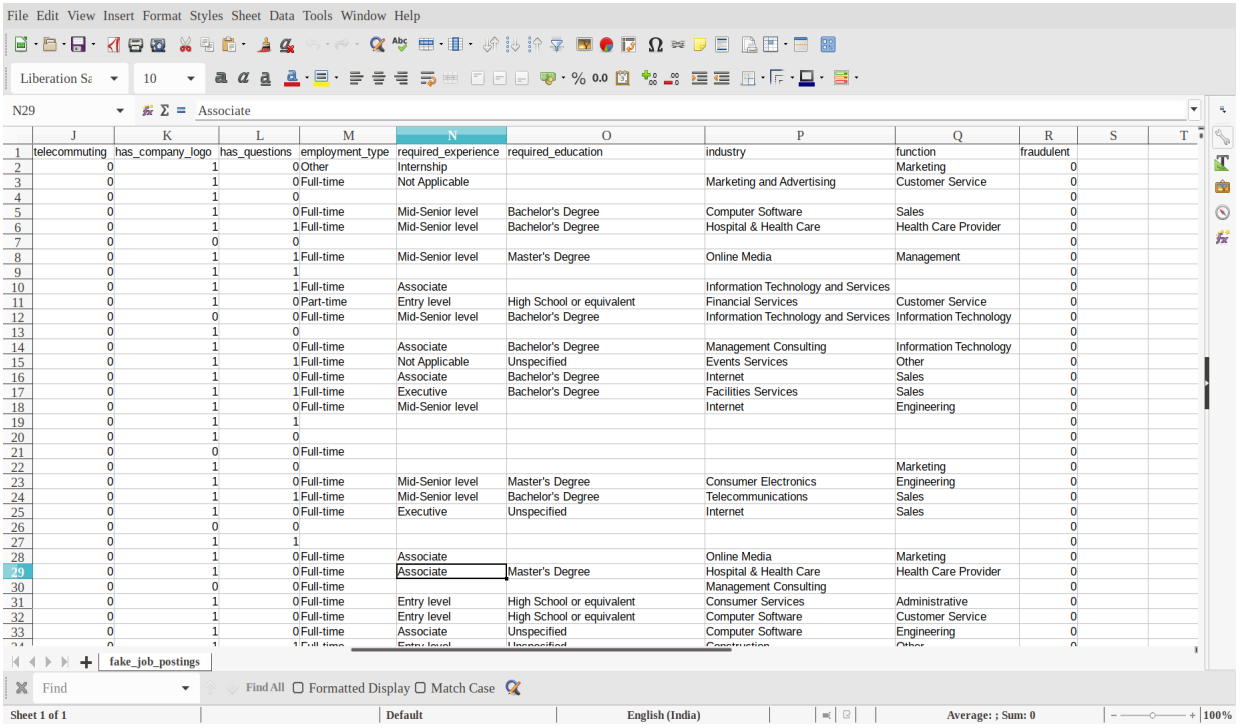


Fig 1.2.1.8: Sample dataset of benefits attribute



	J	K	L	M	N	O	P	Q	R	S	T
1	telecommuting	has_company_logo	has_questions	employment_type	required_experience	required_education	industry	function	fraudulent		
2	0	1	0	Other	Internship			Marketing	0		
3	0	1	0	Full-time	Not Applicable		Marketing and Advertising	Customer Service	0		
4	0	1	0						0		
5	0	1	0	Full-time	Mid-Senior level	Bachelor's Degree	Computer Software	Sales	0		
6	0	1	1	Full-time	Mid-Senior level	Bachelor's Degree	Hospital & Health Care	Health Care Provider	0		
7	0	0	0						0		
8	0	1	1	Full-time	Mid-Senior level	Master's Degree	Online Media	Management	0		
9	0	1	1						0		
10	0	1	1	Full-time	Associate		Information Technology and Services		0		
11	0	1	0	Part-time	Entry level	High School or equivalent	Financial Services	Customer Service	0		
12	0	0	0	Full-time	Mid-Senior level	Bachelor's Degree	Information Technology and Services	Information Technology	0		
13	0	1	0						0		
14	0	1	0	Full-time	Associate	Bachelor's Degree	Management Consulting	Information Technology	0		
15	0	1	1	Full-time	Not Applicable	Unspecified	Events Services	Other	0		
16	0	1	0	Full-time	Associate	Bachelor's Degree	Internet	Sales	0		
17	0	1	1	Full-time	Executive	Bachelor's Degree	Facilities Services	Sales	0		
18	0	1	0	Full-time	Mid-Senior level		Internet	Engineering	0		
19	0	1	1						0		
20	0	1	0						0		
21	0	0	0	Full-time					0		
22	0	1	0					Marketing	0		
23	0	1	0	Full-time	Mid-Senior level	Master's Degree	Consumer Electronics	Engineering	0		
24	0	1	1	Full-time	Mid-Senior level	Bachelor's Degree	Telecommunications	Sales	0		
25	0	1	0	Full-time	Executive	Unspecified	Internet	Sales	0		
26	0	0	0						0		
27	0	1	1						0		
28	0	1	0	Full-time	Associate		Online Media	Marketing	0		
29	0	1	0	Full-time	Associate	Master's Degree	Hospital & Health Care	Health Care Provider	0		
30	0	0	0	Full-time			Management Consulting		0		
31	0	1	0	Full-time	Entry level	High School or equivalent	Consumer Services	Administrative	0		
32	0	1	0	Full-time	Entry level	High School or equivalent	Computer Software	Customer Service	0		
33	0	1	0	Full-time	Associate	Unspecified	Computer Software	Engineering	0		

Fig 1.2.1.9: Sample dataset of remaining attributes

Data preprocessing :

Three common data pre-processing steps are:

- Formatting :
    - The data selected may not be in a format that is suitable to work with. The data may be in a relational database and to be converted into a flat file, or the data may be in a proprietary file format and to be converted to a relational database or a text file.
- Cleaning :
    - Cleaning data is the removal or fixing of missing data. There may be data instances that are incomplete and do not carry the data needed to address the problem. These instances may need to be removed. Additionally, there may be sensitive information in some of the attributes and these attributes may need to be anonymized or removed from the data entirely.

- Sampling :
    - There may be far more selected data available than is needed to work with. More data can result in much longer running times for algorithms and larger computational and memory requirements. Smaller representative sample of the selected data can be taken that may be much faster for exploring and prototyping solutions before considering the whole dataset.

1.2.2 Related work

According to several studies, Review spam detection, Email Spam detection, Fake news detection have drawn special attention in the domain of Online Fraud Detection. A. Review Spam DetectionPeople often post their reviews online forum regarding the products they purchase. It may guide other purchaser while choosing their products. In this context, spammers can manipulate reviews for gaining profit and hence it is required to develop techniques that detect one of these spam reviews. This can be implemented by extracting features from the reviews by extracting features using Natural Language Processing (NLP). Next, machine learning techniques are applied on these features. Lexicon based approaches may be one alternative to machine learning techniques that use the dictionary or corpus to eliminate spam reviews.

Email Spam Detection

Unwanted bulk mails, belonging to the category of spam emails, often arrive in the user's mailbox. This may lead to unavoidable storage crises as well as bandwidth consumption. To eradicate this problem, Gmail, Yahoo mail and Outlook service providers incorporate spam filters using Neural Networks. While addressing the problem of email spam detection, content based filtering, case based filtering, heuristic based filtering, memory or instance based filtering, adaptive spam filtering approaches are taken into consideration.

Fake News Detection

Fake news in social media characterizes malicious user accounts, echo chamber effects. The fundamental study of fake news detection relies on three perspectives- how fake news is written, how fake news spreads, and how a user is related to fake news. Features related to news content and social context are extracted and machine learning models are imposed to recognize fake news.

1.2.3 The proposed model of System

The target of this study is to detect whether a job post is fraudulent or not. Identifying and eliminating these fake job advertisements will help the jobseekers to concentrate on legitimate job posts only a couple of classifiers are employed such as Naive Bayes Classifier, Decision Tree Classifier, K-nearest Neighbor Classifier, and Random Tree Classifier for classifying job post as fake. It is to be noted that the attribute 'fraudulent' of the dataset is kept as target class for classification purpose. At first, the classifiers are trained using 80% of the entire dataset and later 20% of the entire dataset is used for the prediction purpose. The performance measure metrics such as Accuracy, are used for evaluating the prediction for each of these classifiers. Finally, the classifier that has the best performance with respect to the metrics is chosen as the best candidate model.

1.3 Organization of Project

The technique which is developed is taking input as a job_id and compares the input from the label encoded dataset. If the input matches, then it predicts using Random Forest Classifier and displays the result.

We have four modules in our project.

- Data Collection
- Data Pre-Processing
- Apply Algorithm
- Evaluation

2. THEORETICAL ANALYSIS OF THE PROPOSED PROJECT

2.1 Requirements Gathering

2.1.1 Software Requirements

Domain : Machine Learning

Programming Language : Python 3.6

Dataset : fake_job_postings.csv

Packages : Numpy, Pandas, Matplotlib, Scikit-learn, Seaborn, Tkinter

Tool : Jupyter Notebook, Google Colab

2.1.2 Hardware Requirements

Operating System : Windows 7 Ultimate 32 bit / Windows XP/ Windows 10

Processor : Intel Processor

CPU Speed : 2.30 GHz

Memory : 2 GB (RAM)

2.2 Technologies Description

Machine Learning

Machine learning is a method of data analysis that automates analytical model building. It is a branch of artificial intelligence based on the idea that systems can learn from data, identify patterns and make decisions with minimal human intervention. Machine learning focuses on the development of computer programs that can access data and use it to learn for themselves.

The process of learning begins with observations or data in order to look for patterns in data and make better decisions in the future based on the examples that are provided. The primary aim is to allow the computers to learn automatically without human intervention or assistance and adjust actions accordingly.

Machine learning algorithms are often categorized as supervised or unsupervised.

- Supervised machine learning algorithms can apply what has been learned in the past to new data using labeled examples to predict future events.
- Unsupervised machine learning algorithms are used when the information used to train is neither classified nor labeled.

Algorithms used in our project are :

- Logistic Regression
- Random Forest Classifier
- Support Vector Machine
- Decision Tree
- K-Nearest Neighbours
- Naive-Bayes

Single Classifier based Prediction

Classifiers are trained for predicting the unknown test cases. The following classifiers are used while detecting fake job posts.

Naive Bayes Classifier :

The Naive Bayes classifier is a supervised classification tool that exploits the concept of Bayes Theorem of Conditional Probability. The decision made by this classifier is quite effective in practice even if its probability estimates are inaccurate. This classifier obtains a very promising result in the following scenario- when the features are independent or features are completely functionally dependent. The accuracy of this classifier is not related to feature dependencies; rather it is the amount of information loss of the class due to the independence assumption is needed to predict the accuracy. In Gaussian Naive Bayes, continuous values associated with each feature are assumed to be distributed according to a **Gaussian distribution**. A Gaussian distribution is also called Normal distribution. When plotted, it gives a bell shaped curve which is symmetric about the mean of the feature values

Decision Tree Classifier :

A Decision Tree (DT) is a classifier that exemplifies the use of tree-like structure. It gains knowledge on classification. Each target class is denoted as a leaf node of DT and non-leaf nodes of DT are used as a decision node that indicates a certain test. The outcomes of those tests are identified by either of the branches of that decision node. Starting from the beginning at the root this tree goes through it until a leaf node is reached. It is the way of obtaining classification results from a decision tree. Decision tree learning is an approach that has been applied to spam filtering. This can be useful

for forecasting the goal based on some criterion by implementing and training this model.

Important Terminology related to Decision Trees :

1. **Root Node:** It represents the entire population or sample and this further gets divided into two or more homogeneous sets.
2. **Splitting:** It is a process of dividing a node into two or more sub-nodes.
3. **Decision Node:** When a sub-node splits into further sub-nodes, then it is called the decision node.
4. **Leaf / Terminal Node:** Nodes that do not split are called Leaf or Terminal nodes.
5. **Pruning:** When we remove sub-nodes of a decision node, this process is called pruning. You can say the opposite process of splitting.
6. **Branch / Sub-Tree:** A subsection of the entire tree is called branch or sub-tree.
7. **Parent and Child Node:** A node, which is divided into sub-nodes is called a parent node of sub-nodes whereas sub-nodes are the child of a parent node.

Support Vector Machine :

"Support Vector Machine" (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. However, it is mostly used in classification problems. In the SVM algorithm, we plot each data item as a point in n-dimensional space (where n is the number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiates the two classes very well (look at the below snapshot).

The advantages of support vector machines are:

- Effective in high dimensional spaces.
- Still effective in cases where the number of dimensions is greater than the number of samples.
- Uses a subset of training points in the decision function (called support vectors), so it is also memory efficient.

- Versatile: different Kernel functions can be specified for the decision function. Common kernels are provided, but it is also possible to specify custom kernels.

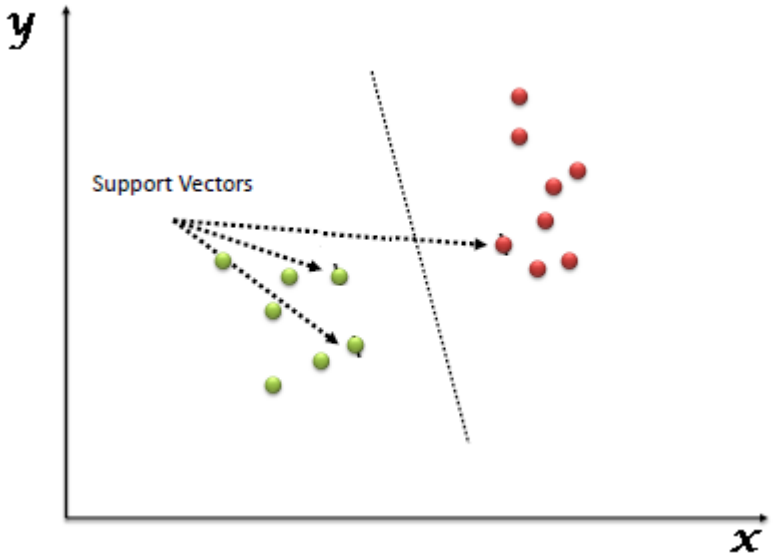


Fig 2.2.1: SVM

Support Vectors are simply the coordinates of individual observation. The SVM classifier is a frontier which best segregates the two classes (hyper-plane/ line).

Logistic Regression : Logistic regression is a supervised learning classification algorithm used to predict the probability of a target variable. The nature of target or dependent variable is dichotomous, which means there would be only two possible classes.

In simple words, the dependent variable is binary in nature having data coded as either 1 (stands for success/yes) or 0 (stands for failure/no).

Mathematically, a logistic regression model predicts P(Y=1) as a function of X. It is one of the simplest ML algorithms that can be used for various classification problems such as spam detection, Diabetes prediction, cancer detection etc.

Types of Logistic Regression

- **Binary or Binomial:** In such a kind of classification, a dependent variable will have only two possible types either 1 and 0

- **Multinomial:** In such a kind of classification, dependent variable can have 3 or more possible unordered types or the types having no quantitative significance.
- **Ordinal:** In such a kind of classification, dependent variable can have 3 or more possible ordered types or the types having a quantitative significance

K-Nearest Neighbor(KNN) : K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique.

- The 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.
- The K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.
- K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data.
- It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.
- The KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.

Ensemble Approach based Classifiers

Ensemble approach facilitates several machine learning algorithms to perform together to obtain higher accuracy of the entire system. Random forest (RF) exploits the concept of ensemble learning approach and regression technique applicable for classification based problems. This classifier assimilates several tree-like classifiers which are applied on various sub-samples of the dataset and each tree casts its vote to the most appropriate class for the input. Boosting is an efficient technique where several unstable learners are assimilated into a single learner in order to improve

accuracy of classification. Boosting technique applies the classification algorithm to the reweighted versions of the training data and chooses the weighted majority vote of the sequence of classifiers. AdaBoost is a good example of boosting technique that produces improved output even when the performance of the weak learners is inadequate. Boosting algorithms are quite efficient in solving spam filtration problems. Gradient boosting algorithm is another boosting technique based classifier that exploits the concept of decision tree. It also minimizes the prediction loss.

Random Forest Algorithm :

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

Python

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently whereas other languages use punctuation, and it has fewer syntactic constructions than other languages.

- **Python is Interpreted**: Python is processed at runtime by the interpreter. Compilation is not needed before executing the program. This is similar to PERL and PHP.
- **Python is Interactive**: You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- **Python is Object-Oriented**: Python supports Object-Oriented style or technique of programming that encapsulates code within objects.

- **Python is a Beginner's Language**: Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

Python's features include:

- **Easy-to-learn**: Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
- **Easy-to-read**: Python code is more clearly defined and visible to the eyes.
- **Easy-to-maintain**: Python's source code is fairly easy-to-maintain.A broad standard library: Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- **Interactive Mode**: Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- **Portable**: Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
- **Extendable**: You can add low-level modules to the Python interpreter.
  These modules enable programmers to add to or customize their tools to be more efficient.
- **Databases**: Python provides interfaces to all major commercial databases.
- **GUI Programming**: Python supports GUI applications that can be created and ported to many system calls, libraries, and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
- **Scalable**: Python provides a better structure and support for large programs than shell scripting.

Jupyter Notebook

The notebook extends the console-based approach to interactive computing in a qualitatively new direction, providing a web-based application suitable for capturing the whole computation process: developing, documenting, and executing code, as well as communicating the results. The Jupyter notebook combines two components:

- Web application
- Notebook documents

A web application: a browser-based tool for interactive authoring of documents which combine explanatory text, mathematics, computations and their rich media output.
Notebook documents: a representation of all content visible in the web application, including inputs and outputs of the computations, explanatory text, mathematics, images, and rich media representations of objects.

Numpy

NumPy, which stands for Numerical Python, is a library consisting of multidimensional array objects and a collection of routines for processing those arrays. Using NumPy, mathematical and logical operations on arrays can be performed. Arbitrary data-types can be defined using Numpy which allows Numpy to seamlessly and speedily integrate with a wide variety of databases. It also discusses the various array functions, types of indexing, etc. NumPy is the basic library for scientific computations in Python.Understanding NumPy is the first major step in the journey of machine learning and deep learning. In order to import numpy the following command is used:

import numpy as np

Pandas

Pandas is a popular Python package for data science, and with good reason: it offers powerful, expressive and flexible data structures that make data manipulation and analysis easy, among many other things. Pandas is an open source high-performance, easy-to-use library providing data structures, such as dataframes, and data analysis tools like the visualization tools.The DataFrame is one of these structures. In order to import Pandas the following command is used :

import pandas as pd

Matplotlib

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. You can generate plots, histograms, power spectra, bar charts, error charts, scatter plots, etc., with just a few lines of code. For

simple plotting the pyplot module provides a MATLAB-like interface, particularly when combined with IPython.It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK.Matplotlib is low level, provides lots of freedom.Python offers multiple great graphing libraries that come packed with lots of different features.In order to import Matplotlib the following command is used :

import matplotlib.pyplot as plt

Scikit - learn

Scikit-learn is an open source machine learning library that supports supervised and unsupervised learning. It uses simple and efficient tools for predictive data analysis.

Scikit-learn is accessible to everybody, and reusable in various contexts. It is built on NumPy, SciPy, and matplotlib. It is also an open source and commercially usable - BSD license

It can be used for :

- Classification
- Regression
- Clustering
- Dimensionality reduction
- Model Selection
- Preprocessing

Seaborn

Seaborn is a data visualization library for Python that runs on top of the popular Matplotlib data visualization library, although it provides a simple interface and aesthetically better-looking plots.

Seaborn requires that Matplotlib is installed first. Seaborn plotting functions expect data to be provided as Pandas DataFrames.

Seaborn can display the following plots :

- **Line Plots**:The relationship between x and y can be shown for different subsets of the data using the hue, size, and style parameters. These parameters control what visual semantics are used to identify the different subsets.
- **Bar Chart Plots:**Plotting a Bar Plot in Seaborn is as easy as calling the barplot() function on the sns instance, and passing in the categorical and continuous variables that we'd like to visualize:
- **Histogram Plots:**A histogram is a classic visualization tool that represents the distribution of one or more variables by counting the number of observations that fall within discrete bins.
- **Box and Whisker Plots:**The seaborn boxplot is a very basic plot Boxplots are used to visualize distributions. Thats very useful when you want to compare data between two groups. Sometimes a boxplot is named a box-and-whisker plot.
- **Scatter Plots:**The scatterplot is a plot with many data points. It is one of the many plots seaborn can create.Seaborn can create this plot with the scatterplot() method. The data points are passed with the parameter data. The parameters x and y are the labels of the plot.

Tkinter

Tkinter is the inbuilt python module that is used to create GUI applications. It is one of the most commonly used modules for creating GUI applications in Python. As it is simple and easy to work with, one need not worry about the installation of the Tkinter module separately as it comes with Python already. It gives an object-oriented interface to the Tk GUI toolkit.

Some other Python Libraries available for creating our own GUI applications are Kivy, Python, Qt wxPython. Among all, Tkinter is most widely used. Graphical User Interface(GUI) is a form of user interface which allows users to interact with computers through visual indicators using items such as icons, menus, windows, etc.

It has advantages over the Command Line Interface(CLI) where users interact with computers by writing commands using keyboard only and whose usage is more difficult than GUI. Tkinter is the inbuilt python module that is used to create GUI applications. It is one of the most commonly used modules for creating GUI applications in Python as it is simple and easy to work with. You don't need to worry about the installation of the Tkinter module separately as it comes with Python already. It gives an object-oriented interface to the Tk GUI toolkit. Widgets in Tkinter are the elements of GUI application which provides various controls (such as Labels, Buttons, ComboBoxes, CheckBoxes, MenuBars, RadioButtons and many more) to users to interact with the application.

3. DESIGN

3.1 Introduction

Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm and area of application. Design is the first step in the development phase for any engineered product or system. The designer's goal is to produce a model or representation of an entity that will later be built. Once system requirements have been specified and analyzed, system design is the first of the three technical activities -design, code and test that is required to build and verify software.

The importance can be stated with a single word "Quality". Design is the place where quality is fostered in software development. Design provides us with representations of software that can assess quality. Design is the only way that we can accurately translate a customer's view into a finished software product or system. Software design serves as a foundation for all the software engineering steps that follow. Without a strong design we risk building an unstable system – one that will be difficult to test, one whose quality cannot be assessed until the last stage.

During design, progressive refinement of data structure, program structure, and procedural details are developed, reviewed and documented. System design can be viewed from either a technical or project management perspective. From the technical point of view, design consists of four activities – architectural design, data structure design, interface design and procedural design.

3.2 Architecture Diagram

Web applications are by nature distributed applications, meaning that they are programs that run on more than one computer and communicate through a network or server. Specifically, web applications are accessed with a web browser and are popular because of the ease of using the browser as a user client. For the enterprise, software on potentially thousands of client computers is a key reason for their popularity. Web applications are used for web mail, online retail sales, discussion boards, weblogs, online banking, and more. One web application can be accessed and used by millions of people.

Like desktop applications, web applications are made up of many parts and often contain mini programs and some of which have user interfaces. In addition, web applications frequently require an additional markup or scripting language, such as HTML, CSS, or JavaScript programming language. Also, many applications use only the Python programming language, which is ideal because of its versatility.

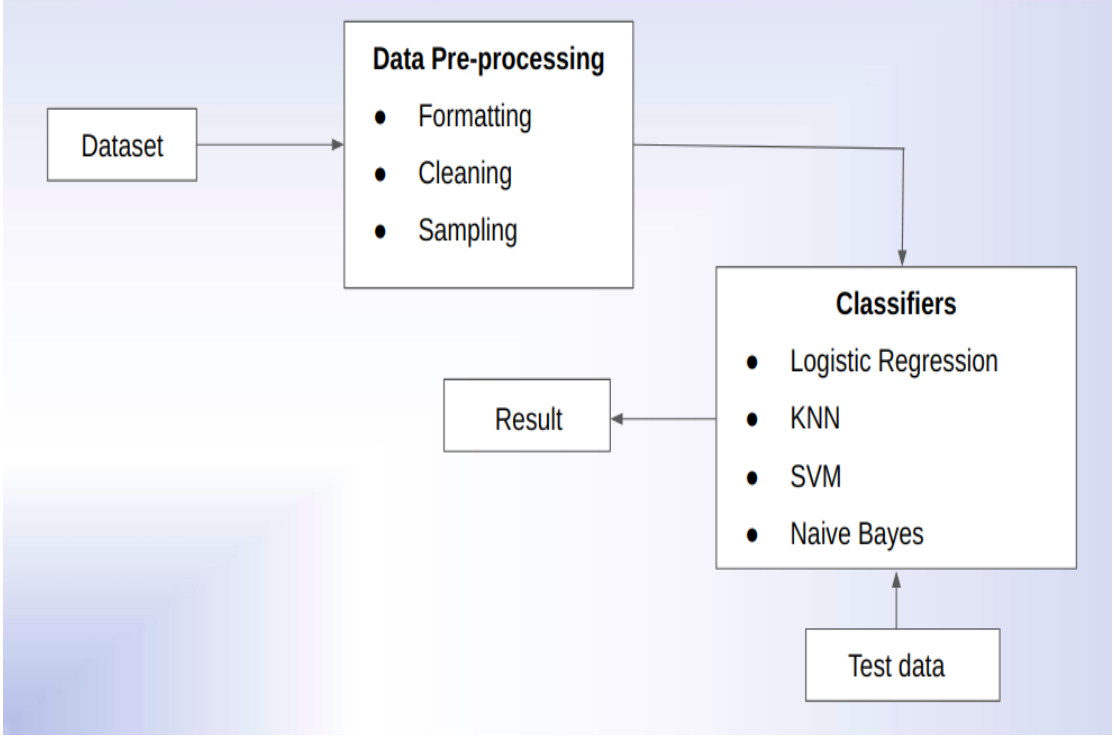


Fig 3.2: Architecture Diagram

3.3 UML Diagrams

UML stands for Unified Modeling Language. It's a rich language to model software solutions, application structures, system behavior and business processes. UML is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems. UML was created by the Object Management Group (OMG) and UML 1.0 specification draft was proposed to the OMG in January 1997. It was initially started to capture the behavior of complex software and non-software systems and now it has become an OMG standard.

List of UML Diagrams:

1. Use case Diagram
2. Sequence diagrams
3. Activity Diagram
4. Class Diagram

3.3.1 Use Case Diagram

To model a system, the most important aspect is to capture the dynamic behavior. Dynamic Behavior means the behavior of the system when it is running/operating. Only static behavior is not sufficient to model a system; rather dynamic behavior is more important than static behavior. In UML, there are five diagrams available to model the dynamic nature and use case diagrams are one of them. Now as we have to discuss that the use case diagram is dynamic in nature, there should be some internal or external factors for making the interaction.

These internal and external agents are known as actors. Use case diagrams consist of actors, use cases and their relationships. The diagram is used to model the system subsystem of an application. A single use case diagram captures a particular functionality of a system.

The purpose of a use case diagram is to capture the dynamic aspect of a system. However, this definition is too generic to describe the purpose, as other four diagrams (activity, sequence, collaboration, and Statechart) also have the same purpose. We will look into some specific purpose, which will distinguish it from the other four diagrams.

Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. Hence, when a system is analyzed to gather its functionalities, use cases are prepared and actors are identified.

Hence to model the entire system, a number of use case diagrams are used.

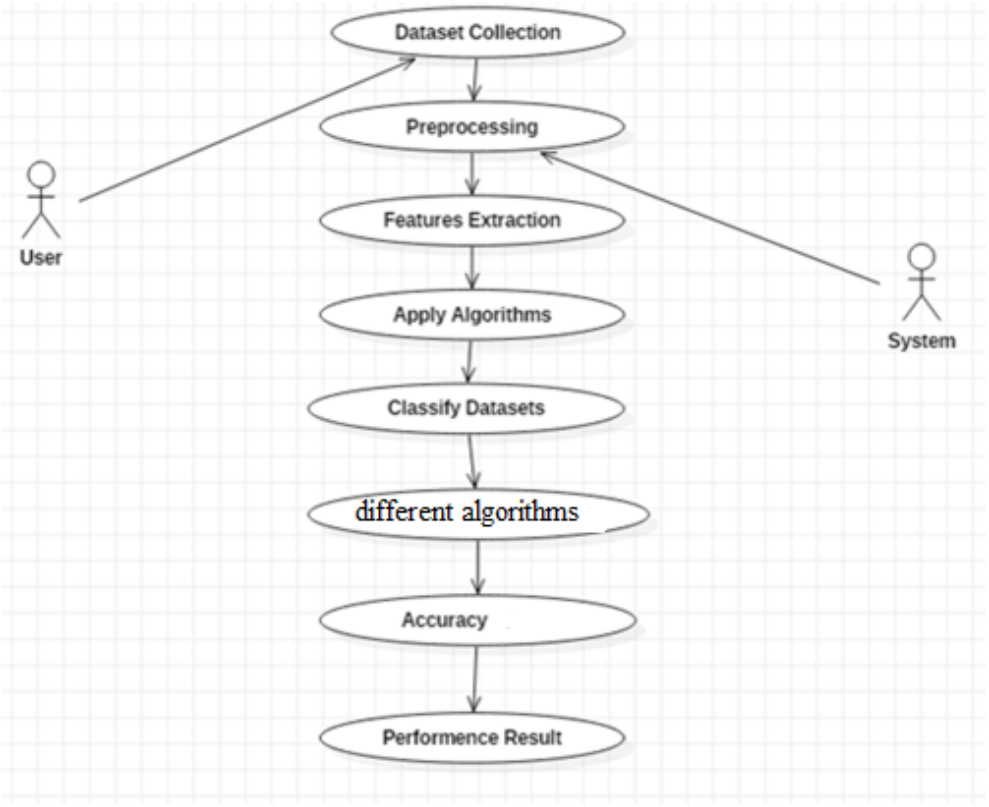


Fig 3.3.1: Use Case Diagram

3.3.2 Sequence Diagram

Sequence Diagrams Represent the objects participating in the interaction horizontally and time vertically. A Use Case is a kind of behavioral classifier that represents a declaration of an offered behavior. Each use case specifies some behavior, possibly including variants that the subject can perform in collaboration with one or more actors. Use cases define the offered behavior of the subject without reference to its internal structure. These behaviors, involving interactions between the actor and the subject, may result in changes to the state of the subject and communications with its environment. A use case can include possible variations of its basic behavior, including exceptional behavior and error handling.

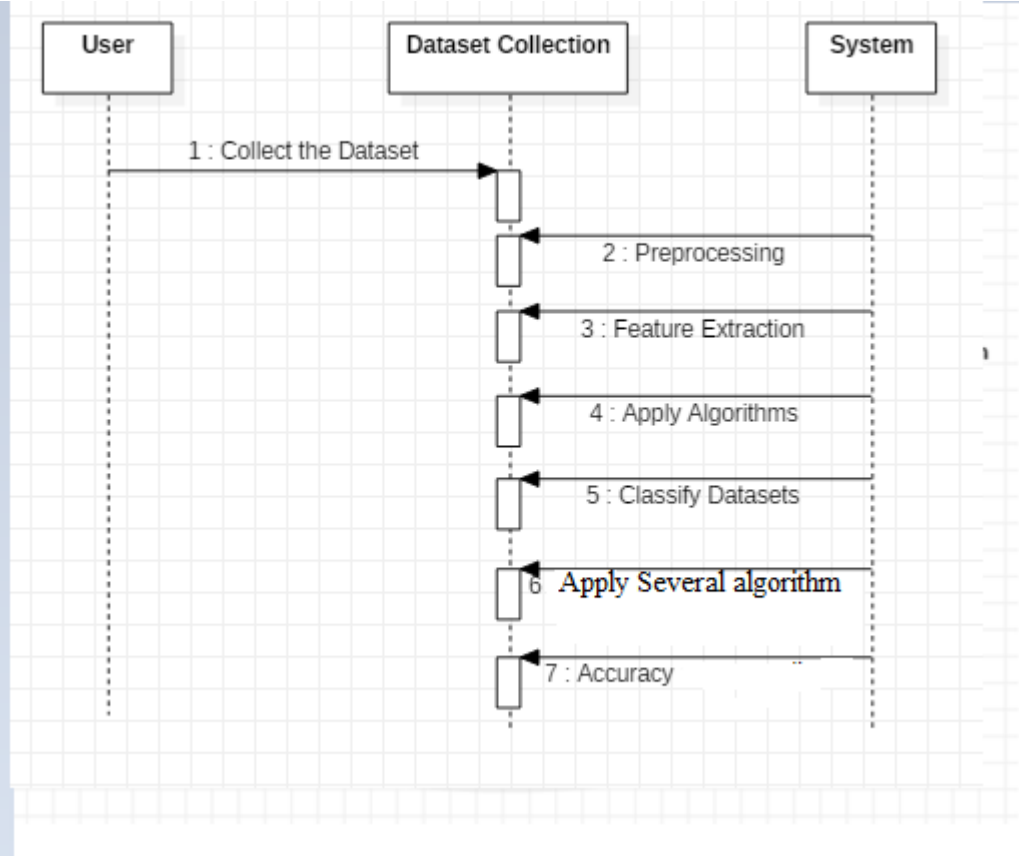


Fig 3.3.2: Sequence Diagram

3.3.3 Activity Diagram

Activity diagrams are graphical representations of Workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

The basic purpose of activity diagrams is similar to the other four diagrams. It captures the dynamic behavior of the system. Other four diagrams are used to show the message flow from one object to another but the activity diagram is used to show message flow from one activity to another.

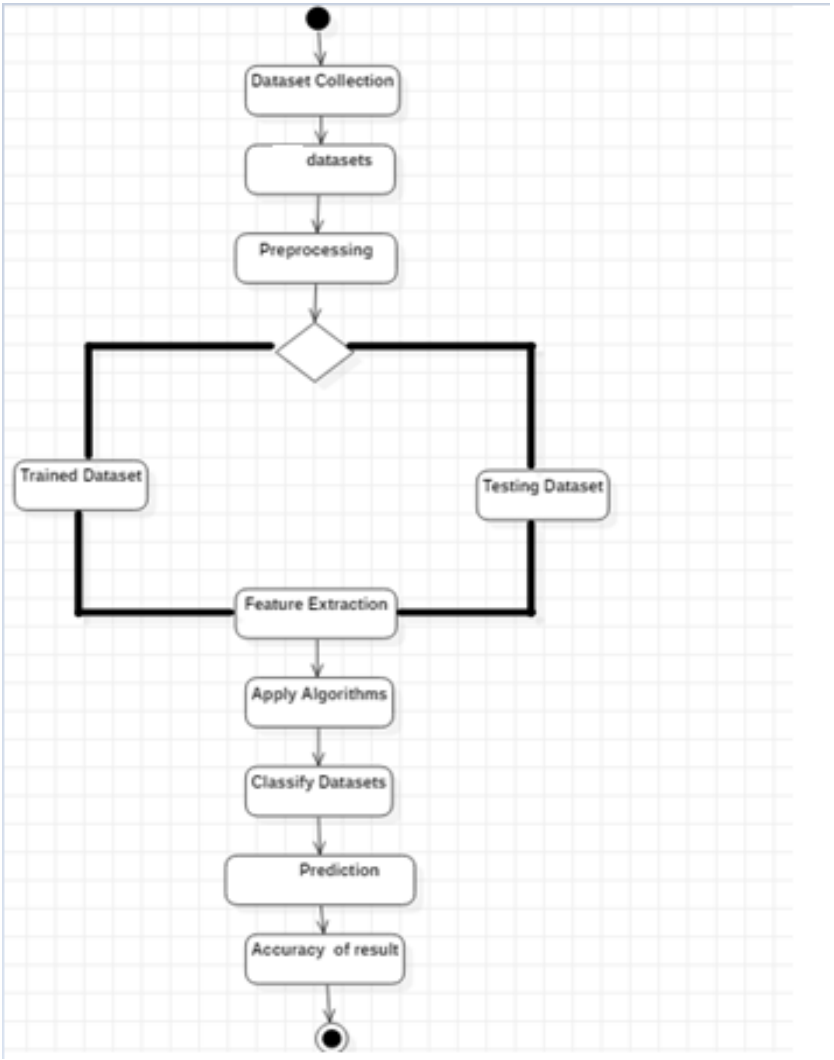


Fig 3.3.3: Activity Diagram

3.3.4 Class Diagram

The class diagram is the main building block of object-oriented modeling. It is used for general conceptual modeling of the system of the application, and for detailed modeling translating the models into programming code. Class diagrams can also be used for data modeling. The classes in a class diagram represent both the main elements, interactions in the application, and the classes to be programmed.

The purpose of the class diagram is to model the static view of an application. Class diagrams are the only diagrams which can be directly mapped with object-oriented languages and thus widely used at the time of construction.

UML diagrams like activity diagram, sequence diagrams can only give the sequence flow of the application, however the class diagram is a bit different. It is the most popular UML diagram in the coder community.

The purpose of the class diagram can be summarized as −

- Analysis and design of the static view of an application.
- Describe responsibilities of a system.
- Base for component and deployment diagrams.
- Forward and reverse engineering.

Fig 3.3.4: Class Diagram

4. IMPLEMENTATION

4.1 Coding

File name : Fake_Job_Recruitment_Detection.ipynb

 # import libraries

```
import numpy as np
import pandas as pd
import seaborn as sns
from sklearn import preprocessing
import matplotlib.pyplot as plt
```

 # load dataset

```
data=pd.read_csv('fake_job_postings.csv')
data.head()

data.shape

data.info()

data.isnull().sum()
```

 # Data Preprocessing

```
data['location'] = data.location.fillna('none')
data['department'] = data.department.fillna('not specified')
data['company_profile'] = data.company_profile.fillna('none')
data['requirements'] = data.requirements.fillna('not specified')
data['employment_type'] = data.employment_type.fillna('not specified')
data['required_experience'] = data.required_experience.fillna('not specified')
data['required_education'] = data.required_education.fillna('not specified')
data['industry'] = data.industry.fillna('not specified')
data['function'] = data.function.fillna('not specified')
```

```
data.drop(['salary_range', 'benefits','telecommuting','has_questions'], axis=1,
inplace=True)

data.isnull().sum()

data.head()

data.columns

# find unique values in an attribute
print('Data set:')
for col_name in data.columns:
    if data[col_name].dtypes == 'object' :
        unique_cat = len(data[col_name].unique())
        print("Feature '{col_name}' has {unique_cat}categories".format(col_name =
col_name, unique_cat = unique_cat))

print()

df = data[['job_id','title', 'location','company_profile', 'requirements',
'employment_type', 'required_experience', 'required_education', 'industry', 'function',
'fraudulent']]

df.isnull().sum()

df_num = df[['fraudulent']]
df_cat = df[['title', 'location','company_profile', 'requirements','employment_type',
      'required_experience', 'required_education', 'industry', 'function']]

# Checking for Outliers in numerical data
plt.figure(figsize=[16,8])
sns.boxplot(data = df_num)
plt.show()
```

```
#Removing Outliers from columns
df_num = df_num[df_num['fraudulent'] < 0.9 ]
plt.figure(figsize=[16,8])
sns.boxplot(data = df_num)
plt.show()

# Visualisation
fig, axes = plt.subplots(ncols=2, figsize=(17, 5), dpi=100)
plt.tight_layout()

df["fraudulent"].value_counts().plot(kind='pie', ax=axes[0], labels=['Real Post (95%)',
'Fake Post (5%)'])
temp = df["fraudulent"].value_counts()
sns.barplot(temp.index, temp, ax=axes[1])

axes[0].set_ylabel(' ')
axes[1].set_ylabel(' ')
axes[1].set_xticklabels(["Real Post (17014) [0's]", "Fake Post (866) [1's]"])

axes[0].set_title('Target Distribution in Dataset', fontsize=13)
axes[1].set_title('Target Count in Dataset', fontsize=13)
plt.show()

cat_cols = ["employment_type", "required_experience", "required_education",]

# Visualizing categorical variable by target
import matplotlib.gridspec as gridspec # to do the grid of plots
grid = gridspec.GridSpec(3, 3, wspace=0.5, hspace=0.5) # The grid of chart
plt.figure(figsize=(15,25)) # size of figure

# loop to get column and the count of plots
for n, col in enumerate(df[cat_cols]):
```

```
    ax = plt.subplot(grid[n]) # feeding the figure of grid
    sns.countplot(x=col, data=df, hue='fraudulent', palette='Set2')

    ax.set_ylabel('Count', fontsize=12) # y axis label
    ax.set_title(f'{col} Distribution by Target', fontsize=15) # title label
    ax.set_xlabel(f'{col} values', fontsize=12) # x axis label

    xlabels = ax.get_xticklabels()
    ylabels = ax.get_yticklabels()

    ax.set_xticklabels(xlabels,  fontsize=10)
    ax.set_yticklabels(ylabels,  fontsize=10)

    plt.legend(fontsize=8)
    plt.xticks(rotation=90)
    total = len(df)
    sizes=[] # Get highest values in y
    for p in ax.patches: # loop to all objects
        height = p.get_height()
        sizes.append(height)
        ax.text(p.get_x()+p.get_width()/2.,
                height + 3,
                '{:1.2f}%'.format(height/total*100),
                ha="center", fontsize=10)
    ax.set_ylim(0, max(sizes) * 1.15) #set y limit based on highest heights
plt.show()

fig,(ax1,ax2)= plt.subplots(ncols=2, figsize=(17, 5), dpi=100)

length=df[df["fraudulent"]==1]['requirements'].str.len()
ax1.hist(length,bins = 20,color='orangered')
ax1.set_title('Fake Post')
```

```
length=df[df["fraudulent"]==0]['requirements'].str.len()
ax2.hist(length, bins = 20)
ax2.set_title('Real Post')
fig.suptitle('Characters in description')
plt.show()

fig,(ax1,ax2)= plt.subplots(ncols=2, figsize=(17, 5), dpi=100)

num=df[df["fraudulent"]==1]['company_profile'].str.split().map(lambda x: len(x))
ax1.hist(num,bins = 20,color='orangered')
ax1.set_title('Fake Post')

num=df[df["fraudulent"]==0]['company_profile'].str.split().map(lambda x: len(x))
ax2.hist(num, bins = 20)
ax2.set_title('Real Post')
fig.suptitle('Words in company profile')
plt.show()

fraud = df[df['fraudulent']== 1]
fraud.shape

fraud

not_fraud = df[df['fraudulent']== 0]
not_fraud.shape

not_fraud

df = fraud.append(not_fraud)
df

# Encoding text data
from sklearn.preprocessing import LabelEncoder
```

```
le = LabelEncoder()

df['title'] = le.fit_transform(df['title'])
df['location'] = le.fit_transform(df['location'])
df['company_profile'] = le.fit_transform(df['company_profile'])
df['requirements'] = le.fit_transform(df['requirements'])
df['employment_type'] = le.fit_transform(df['employment_type'])
df['required_experience'] = le.fit_transform(df['required_experience'])
df['required_education'] = le.fit_transform(df['required_education'])
df['industry'] = le.fit_transform(df['industry'])
df['function'] = le.fit_transform(df['function'])

df.reset_index(inplace = True, drop = True)
df

from sklearn.model_selection import train_test_split

X = df[['job_id', 'title', 'location', 'company_profile', 'requirements',
'employment_type','required_experience', 'required_education', 'industry',
'function']].values
Y = df[['fraudulent']].values
X_train, X_test, Y_train, Y_test = train_test_split(X, Y)
```

#import libraries

```
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.svm import SVC
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import GaussianNB
import sklearn.metrics as metrics
from sklearn.metrics import accuracy_score
```

```
import warnings
warnings.filterwarnings('ignore')
```

Logistic Regression

```
clf1=LogisticRegression()
clf1.fit(X_train, Y_train)
preds=clf1.predict(X_test)
print('accuracy with Logistic Regression:',accuracy_score(Y_test, preds), '%')
```

Random Forest

```
clf2=RandomForestClassifier()
clf2.fit(X_train, Y_train)
preds=clf2.predict(X_test)
print('accuracy with Random Forest:',accuracy_score(Y_test, preds), '%')
```

Support Vector Machine

```
clf3=SVC()
clf3.fit(X_train, Y_train)
preds=clf3.predict(X_test)
print('accuracy with Support Vector Machine:',accuracy_score(Y_test, preds), '%')
```

Decision Tree

```
clf4=DecisionTreeClassifier()
clf4.fit(X_train, Y_train)
preds=clf4.predict(X_test)
print('accuracy with Decision Tree:',accuracy_score(Y_test, preds), '%')
```

K-Nearest Neighbors

```
clf5=KNeighborsClassifier()
clf5.fit(X_train, Y_train)
preds=clf5.predict(X_test)
print('accuracy with K-Nearest Neighbors :',accuracy_score(Y_test, preds), '%')
```

```
### Naive Bayes
clf6=GaussianNB()
clf6.fit(X_train, Y_train)
preds=clf6.predict(X_test)
print('accuracy with Naive Bayes:',accuracy_score(Y_test, preds), '%')

# User Interface
from tkinter import *
window = Tk()
window.title("Fake job recruitment detection")
window.geometry('500x200')

lbl = Label(window, text="Enter job id", width = 10)
lbl.grid(column=0, row=0, padx=(0, 50), pady = 10)

txt = Entry(window,width=20)
txt.grid(column=1, row=0, pady=10)

result = Label(window, text='')
result.grid(column=1, row=2, pady=10)

def check() :
    job_id = txt.get()
    if not job_id :
        result.configure(text="Please enter Id")
    else :
        if job_id.isdigit():
            detect(int(job_id))
        else :
            result.configure(text="Please enter a number")

def detect(job_id):
    s = ''
```

```
    if (job_id < 1 or job_id > 17880) :
        result.configure(text="Please enter valid Id [1 - 17880]")
    else :
        s = "Posted job with job id '" + str(job_id) + "' is "

        test_df = df.loc[df["job_id"] == job_id]
        del test_df['fraudulent']
        test_df.reset_index(inplace = True, drop = True)

        print("Logistic Regression : " + str(clf1.predict(test_df)[0]))
        predicted = int(clf2.predict(test_df)[0])
        print("Random Forest : " + str(predicted))
        print("Support Vector Machine : " + str(clf3.predict(test_df)[0]))
        print("Decision Tree : " + str(clf4.predict(test_df)[0]))
        print("K-Nearest Neighbors : " + str(clf5.predict(test_df)[0]))
        print("Naive Bayes : " + str(clf6.predict(test_df)[0]))
        print()

        s +=  "real" if predicted == 0 else "fake"
        result.configure(text=s)

btn = Button(window, text="Detect", command=check)
btn.grid(column=1, row=1,pady=10)
window.mainloop()
```

5. TESTING

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. The increasing visibility of software as a system element and attendant costs associated with a software failure are motivating factors for we planned, through testing. Testing is the process of executing a program with the intent of finding an error. The design of tests for software and other engineered products can be as challenging as the initial design of the product itself.

There are basically two types of testing approaches.

One is Black-Box testing – the specified function that a product has been designed to perform, tests can be conducted that demonstrate each function is fully operated.

The other is White-Box testing – knowing the internal workings of the product ,tests can be conducted to ensure that the internal operation of the product performs according to specifications and all internal components have been adequately exercised.

White box and Black box testing methods have been used to test this package. The entire loop constructs have been tested for their boundary and intermediate conditions. The test data was designed with a view to check for all the conditions and logical decisions. Error handling has been taken care of by the use of exception handlers.

5.1 Testing Strategies

Testing is a set of activities that can be planned in advance and conducted systematically. A strategy for software testing must accommodate low-level tests that are necessary to verify that a small source code segment has been correctly implemented as well as high-level tests that validate major system functions against customer requirements.

Software testing is one element of verification and validation. Verification refers to the set of activities that ensure that software correctly implements a specific function. Validation refers to a different set of activities that ensure that the software that has been built is traceable to customer requirements.

The main objective of software is testing to uncover errors. To fulfill this objective, a series of test steps unit, integration, validation and system tests are planned and executed. Each test step is accomplished through a series of systematic test techniques that assist in the design of test cases. With each testing step, the level of abstraction with which software is considered is broadened.

Testing is the only way to assure the quality of software and it is an umbrella activity rather than a separate phase. This is an activity to be performed in parallel with the software effort and one that consists of its own phases of analysis, design, implementation, execution and maintenance.

5.2 Types of Testing

Unit testing:

This testing method considers a module as a single unit and checks the unit at interfaces and communicates with other modules rather than getting into details at statement level. Here the module will be treated as a black box, which will take some input and generate output. Outputs for a given set of input combinations are pre-calculated and are generated by the module.

System testing:

Here all the pre-tested individual modules will be assembled to create the larger system and tests are carried out at system level to make sure that all modules are working in synchrony with each other. This testing methodology helps in making sure that all modules which are running perfectly when checked individually are also running in cohesion with other modules. For this testing we create test cases to check all modules at once and then generate test combinations of test paths throughout the system to make sure that no path is making its way into chaos.

Integrated testing:

Testing is a major quality control measure employed during software development. Its basic function is to detect errors. Sub functions when combined may not produce more than it is desired. Global data structures can represent the problems. Integrated testing is a systematic technique for constructing the program structure while conducting the tests. To uncover errors that are associated with interfacing the

objective is to make unit test modules and build a program structure that has been detected by design. In a non - incremental integration all the modules are combined in advance and the program is tested as a whole. Here errors will appear in an endless loop function. In incremental testing the program is constructed and tested in small segments where the errors are isolated and corrected.

Different incremental integration strategies are top – down integration, bottom – up integration, regression testing.

Regression testing:

Each time a new module is added as a part of integration as the software changes. Regression testing is an actual that helps to ensure changes that do not introduce unintended behavior as additional errors.

Regression testing may be conducted manually by executing a subset of all test cases or using automated capture playback tools enables the software engineer to capture the test case and results for subsequent playback and compression. The regression suit contains different classes of test cases.

A representative sample of tests that will exercise all software functions.

Additional tests that focus on software functions that are likely to be affected by the change.

5.3 Test cases

Unit testing strategy is used in this application for testing.

Test Case Id	Test Scenario	Expected Result	Actual Result	Pass/Fail
TO1	Check whether jupyter notebook is installed	Jupyter notebook should be opened after executing command	As expected	Pass
TO2	Check if all the packages are installed	Error should not be displayed	As expected	Pass
TO3	Check if all the modules are correctly imported	Error should not be displayed	As expected	Pass
TO4	Check for empty input	Warning message should be given	As expected	Pass
TO5	Check for string input	Warning message should be given	As expected	Pass
TO6	Check for out of range input	Warning message should be given	As expected	Pass
TO7	Check whether button is working	Should give result	As expected	Pass
TO8	Check whether getting correct output	It should correctly predict output	As expected	Pass

Fig 5.3: Test cases

5.4 Visualization Screenshots

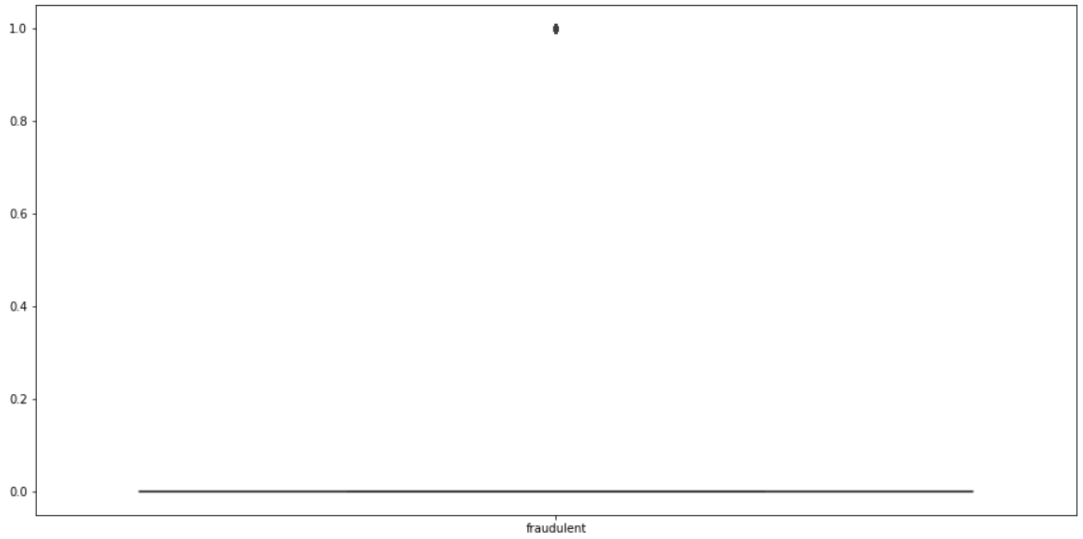


Fig 5.4.1: Outliers in fraudulent attribute

Fig 5.4.2: After removing outliers in fraudulent attribute

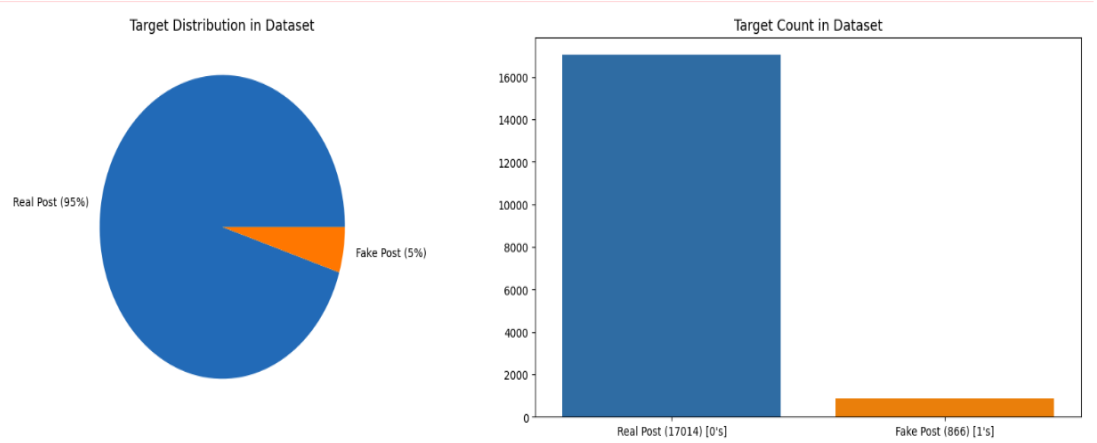


Fig 5.4.3: Fraudulent percentage in dataset

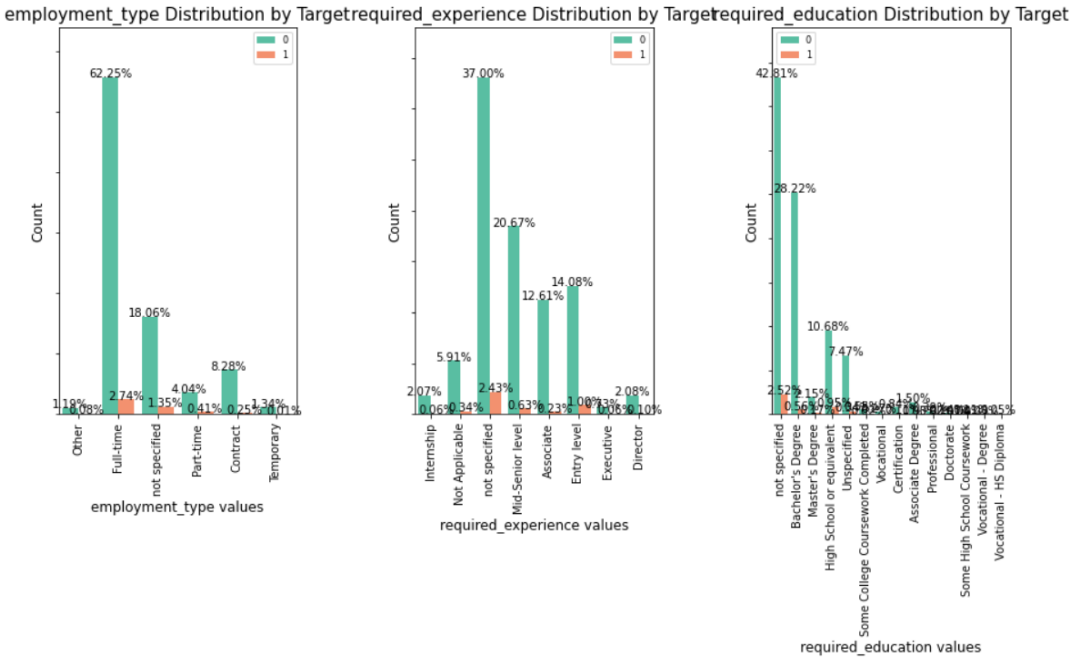


Fig 5.4.4: Visualizing categorical variable by target

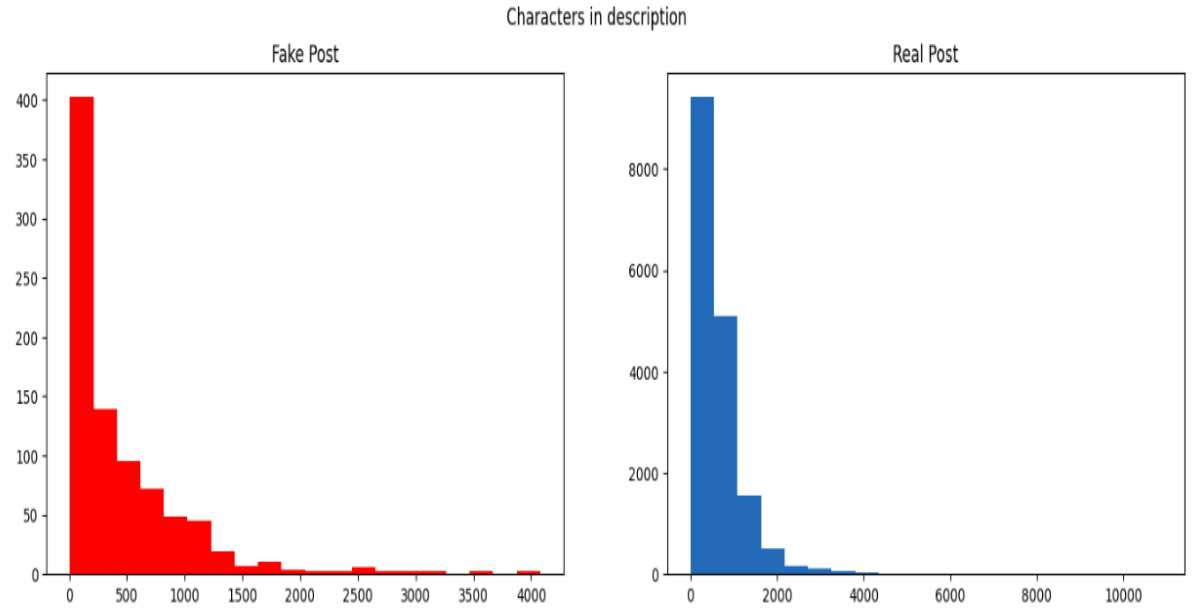


Fig 5.4.5: Comparison on characters in description

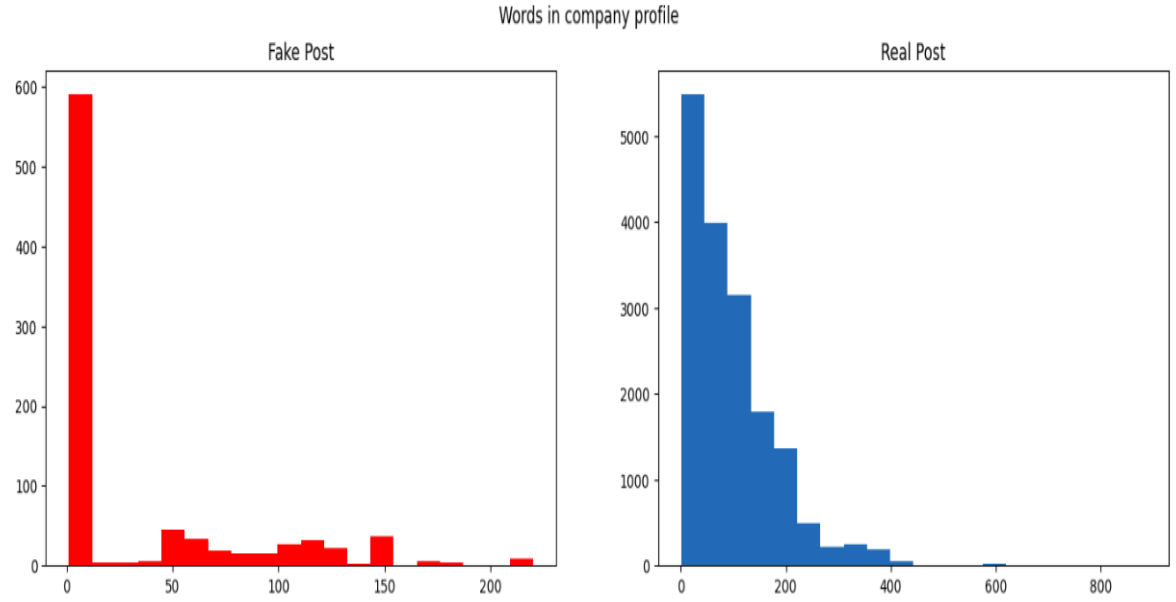


Fig 5.4.6: Comparison on number of words in Company profile

5.5 Input Screenshots

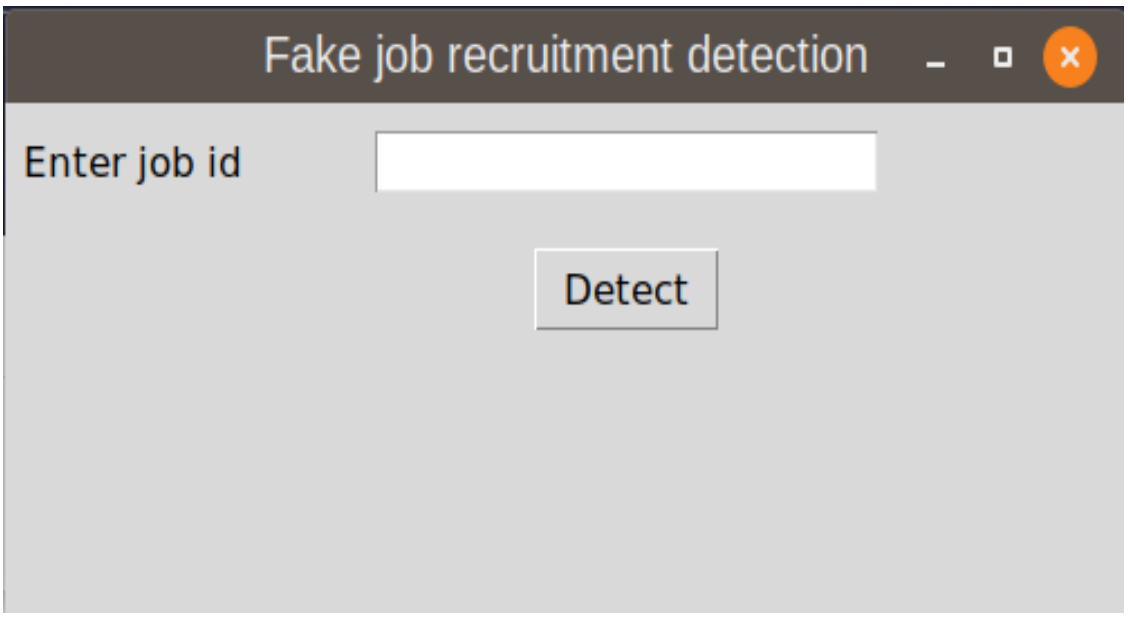


Fig 5.5: Input screen

6. OUTPUT SCREENSHOTS

```
accuracy with Logistic Regression: 0.9250814332247557 %
accuracy with Random Forest: 0.9876221498371336 %
accuracy with Support Vector Machine: 0.9761129207383279 %
accuracy with Decision Tree: 0.9765472312703583 %
accuracy with K-Nearest Neighbors : 0.9454940282301846 %
accuracy with Naive Bayes: 0.9274701411509229 %
```

Fig 6.1: Accuracy of algorithms

```
user@user-Aspire-V3-574: ~
File  Edit  View  Search  Terminal  Help
user@user-Aspire-V3-574:~$ jupyter notebook
[I 15:50:17.243 NotebookApp] Serving notebooks from local directory: /home/user
[I 15:50:17.243 NotebookApp] Jupyter Notebook 6.4.0 is running at:
[I 15:50:17.243 NotebookApp] http://localhost:8888/?token=8d0df54d4980e3d96844d1
f279c822120a63ad2e237ba94d
[I 15:50:17.243 NotebookApp]  or http://127.0.0.1:8888/?token=8d0df54d4980e3d968
44d1f279c822120a63ad2e237ba94d
[I 15:50:17.243 NotebookApp] Use Control-C to stop this server and shut down all
 kernels (twice to skip confirmation).
[C 15:50:17.249 NotebookApp]

    To access the notebook, open this file in a browser:
        file:///home/user/.local/share/jupyter/runtime/nbserver-9868-open.html
    Or copy and paste one of these URLs:
        http://localhost:8888/?token=8d0df54d4980e3d96844d1f279c822120a63ad2e237
ba94d
     or http://127.0.0.1:8888/?token=8d0df54d4980e3d96844d1f279c822120a63ad2e237
ba94d
Opening in existing browser session.
```

Fig 6.2: Command to open jupyter book

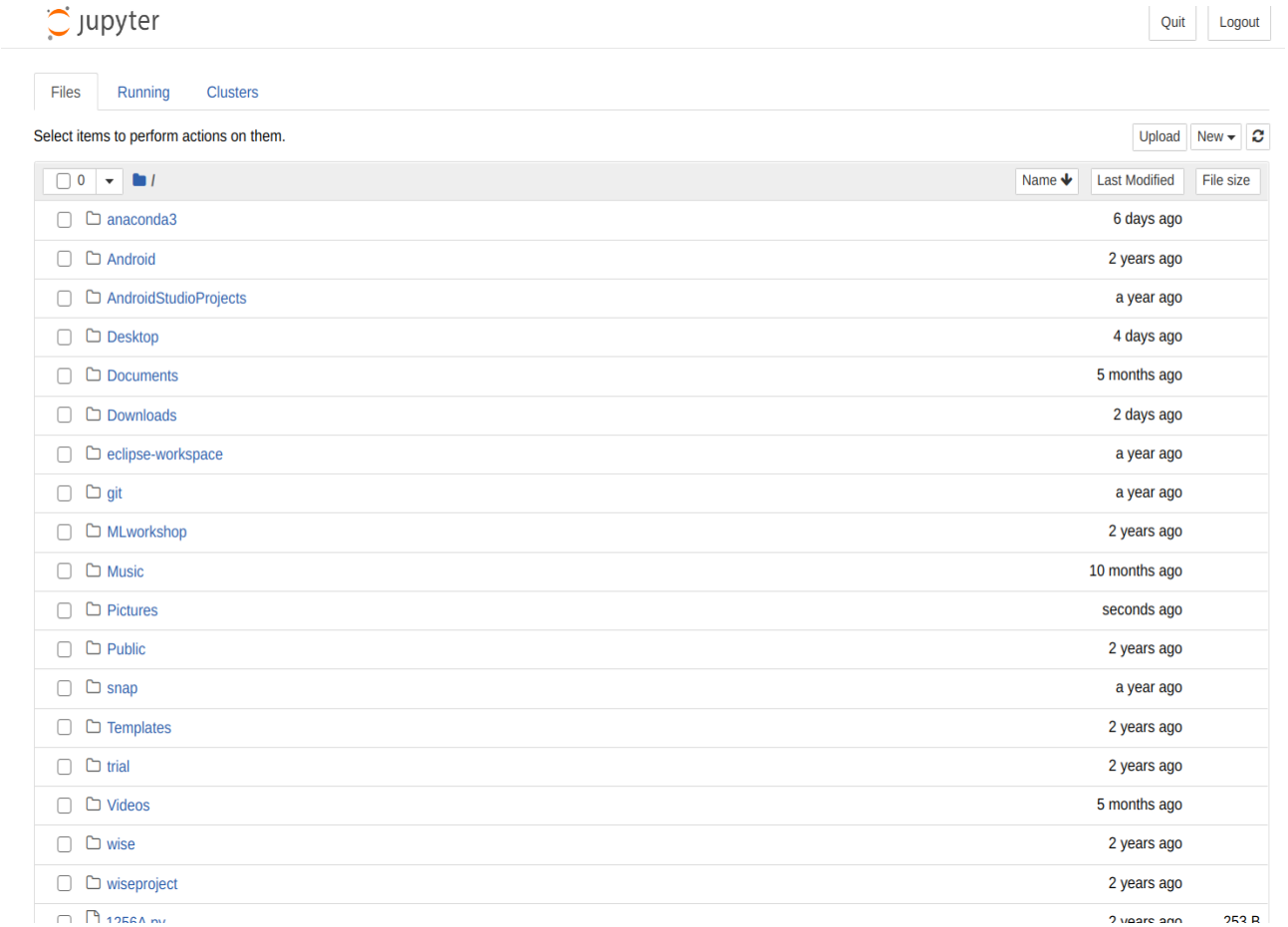


Fig 6.3: Test case showing jupyter notebook has opened

```
import numpy as np
import pandas as pd
import seaborn as sns
from sklearn import preprocessing
import matplotlib.pyplot as plt
```

Fig 6.4: Test case showing there is no error while importing modules and packages

Fig 6.5: If no input is given, "Please enter Id" is displayed

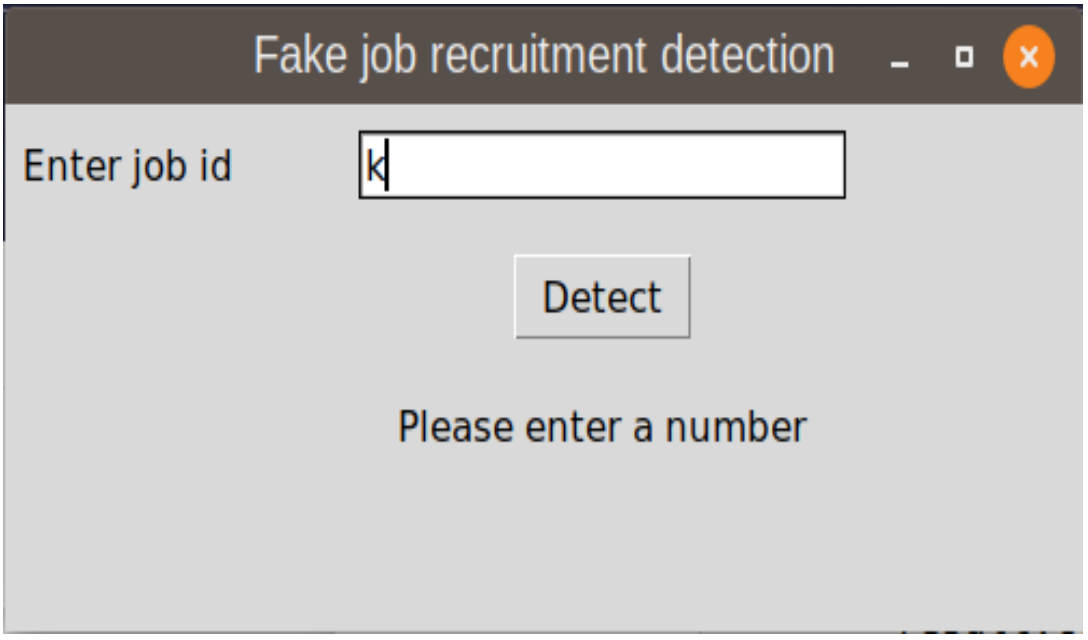


Fig 6.6: If string is given as input, "Please enter a number" is displayed

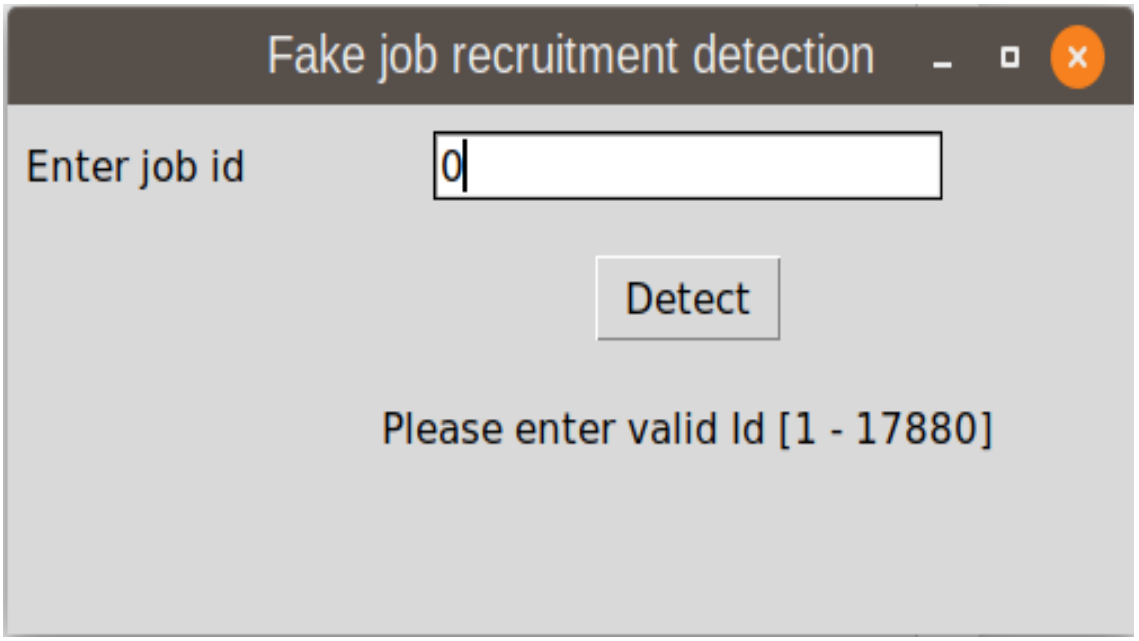


Fig 6.7: If a number out of range is given as input, "Please enter valid Id [1-17880]" is displayed

Fig 6.8: Given input '145' output is predicted

```
Logistic Regression : 0
Random Forest : 1
Support Vector Machine : 0
Decision Tree : 1
K-Nearest Neighbors : 0
Naive Bayes : 0
```

Fig 6.9: Predicted output by all algorithms (0 - real; 1 - fake)

Fig 6.1: Given input '98' output is predicted

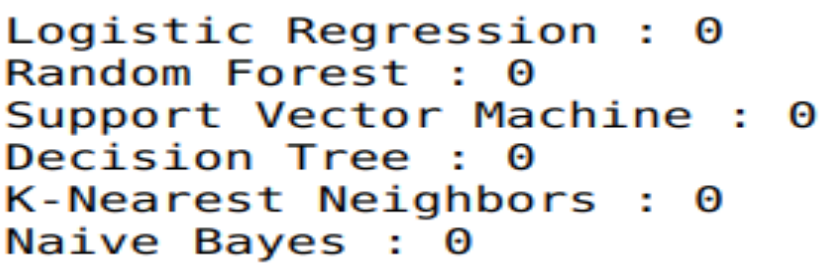


Fig 6.11: Predicted output by all algorithms (0 - real; 1 - fake)

7. CONCLUSION AND FUTURE SCOPE

Fake Job detection will guide job-seekers to get only legitimate offers from companies. For tackling employment scam detection, several machine learning algorithms are proposed as countermeasures in this paper. Supervised mechanism is used to exemplify the use of several classifiers for Fraudulent Job detection. Experimental results indicate that Random Forest classifier outperforms over its peer classification tool. From the proposed approaches highest achieved accuracy is 98.76% which is much higher than the existing methods.

The future enhancement of this application is

- To increase the accuracy using a neural network.
- To make a licensed website.

8. REFERENCES

[1] B. Alghamdi and F. Alharby, ―An Intelligent Model for Online Recruitment Fraud Detection," J. Inf. Secur., vol. 10, no. 03, pp. 155–176, 2019, doi: 10.4236/jis.2019.103009.

[2] I. Rish, ―An Empirical Study of the Naïve Bayes Classifier An empirical study of the naive Bayes classifier,‖ no. January 2001, pp. 41–46, 2014.

[3] D. E. Walters, ―Bayes's Theorem and the Analysis of Binomial Random Variables,‖ Biometrical J., vol. 30, no. 7, pp. 817–825, 1988, doi: 10.1002/bimj.4710300710.

[4] F. Murtagh, ―Multilayer perceptrons for classification and regression,‖ Neurocomputing, vol. 2, no. 5–6, pp. 183–197, 1991, doi: 10.1016/0925-2312(91)90023-5.

[5] P. Cunningham and S. J. Delany, ―K -Nearest Neighbour Classifiers,‖ Mult. Classif. Syst., no. May, pp. 1–17, 2007, doi: 10.1016/S0031-3203(00)00099-6.

[6] H. Sharma and S. Kumar, ―A Survey on Decision Tree Algorithms of Classification in Data Mining,‖ Int. J. Sci. Res., vol. 5, no. 4, pp. 2094–2097, 2016, doi: 10.21275/v5i4.nov162954.

[7] E. G. Dada, J. S. Bassi, H. Chiroma, S. M. Abdulhamid, A. O. Adetunmbi, and O. E. Ajibuwa, "Machine learning for email spam filtering: review, approaches and open research problems,‖ Heliyon, vol. 5, no. 6, 2019, doi: 10.1016/j.heliyon.2019.e01802.

[8] L. Breiman, ―ST4_Method_Random_Forest,‖ Mach. Learn., vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1017/CBO9781107415324.004.

[9] B. Biggio, I. Corona, G. Fumera, G. Giacinto, and F. Roli, ―Bagging classifiers for fighting poisoning attacks in adversarial classification tasks," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 6713 LNCS, pp. 350–359, 2011, doi: 10.1007/978-3-642-21557-5_37.

[10] A. Natekin and A. Knoll, ―Gradient boosting machines, a tutorial,‖ Front. Neurorobot., vol. 7, no. DEC, 2013, doi: 10.3389/fnbot.2013.00021.