Fake E Job Posting Prediction

Submitted By

Ali Raza

(2017-2021)

Department of Computer Science
Khwaja Fareed University of Engineering & Information Technology
Rahim Yar Khan
2021

Fake E Job Posting Prediction

Project
Submitted to

Mr. Saqib Ubaid

Department of Computer Science

In partial fulfilment of the requirements
For the degree of

Bachelor of Science in Computer Science

By

Ali Raza

CS172106

(2017-2021)

Khwaja Fareed University of Engineering & Information Technology
Rahim Yar Khan
2021

DECLARATION

I hereby declare that this project report is based on my original work except for citations and quotations which have been duly acknowledged. I also declare that it has not been previously and concurrently submitted for any other degree or award at Khwaja Fareed University of engineering & Information Technology or other institutions.

Name: Ali Raza

Reg No: CS172106

Signature: _____

APPROVAL FOR SUBMISSION

I certify that this project report entitled **"Fake E Job Posting Prediction"** was prepared by **Ali Raza** has met the required standard for submission in partial fulfilment of the requirements for the award of Bachelor of Science Computer Science (Honours) at Khwaja Fareed University of Engineering & Information Technology.

Approved by:

Supervisor : Mr. Saqib Ubaid

Signature : _____

Date : _____

ACKNOWLEDGEMENT

First of all, I would like to thank my institute, **KFUEIT**. Development and documentation phases of FYP are a great chance of learning and professional development for me.

I gratefully acknowledge the support and patience of my family, professors and friends throughout my studies without them this project report can never be completed.

I am grateful to **Dr. Saleem Ullah**, Head of Computer Science Category, for their support and appreciation. Also, very grateful to **Mr. Saqib Ubaid**, Lecturer of Computer Science Category and Supervisor of my project, for support, guideline and great supervision.

I again thank my course fellows for their good cooperation during the course. Throughout this phase of documentation, I did not only gain a lot of knowledge but more importantly, I also had a great chance to sharpen my skills in a professional working environment.

Above all, to the Almighty ALLAH, for granting knowledge and for all the blessings that He has provided and poured upon me. He has shown His unconditional and pure love by using the people around me who are able to let me feel that I am loved and cared.
I would like to thank everyone who had contributed to the successful completion of this project. I would like to express my gratitude to my Project supervisor, **Mr. Saqib Ubaid** for his invaluable advice, guidance and his enormous patience throughout the development of the research.

In addition, I would also like to express my gratitude to my loving parents and friends who had helped and given me encouragement.

ABSTRACT

To avoid fraudulent posts for job in the internet, an automated online tool using machine learning based classification and techniques is proposed. It helps in detecting fraudulent job posts from an enormous number of posts on internet. There are a lot of job advertisements on the internet, even on the reputed job advertising sites, which never seem fake. But after the selection, the so-called recruiters start asking for the money and the bank details. Many of the candidates fall in their trap and lose a lot of money and the current job sometimes. So, it is better to identify whether a job advertisement posted on the site is real or fake. Identifying it manually is very difficult and almost impossible! To avoid fraudulent posts for job in the internet, an automated online tool (website) using machine learning based classification and techniques is proposed.

Keywords: Fraudulent, Job, Machine learning, Real, Fake, Job advertisement, classification

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1
Introduction

1.1 Introduction

Employment scam is one of the serious issues in recent times addressed in the domain of Online Recruitment Frauds. We are living in unprecedented times due to COVID-19 pandemic hurting economies in every continent. Unemployment rates are increasing every single day with the United States reporting around 26 million people, which is the highest recorded in its long history.

In the latest update to its World Economic Outlook, the IMF (International Monetary Fund) has projected unemployment in Pakistan at 13 percent for 2020 against 7.3 percent in 2019 and 3.9 percent in 2018. 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 a 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, 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.

Sexual harassment, abuse and discrimination in Pakistan’s workplaces, including universities, are pervasive, mostly unreported and ignored by senior managers, a Dawn survey of 300 women found. In response to being asked whether women were made to stay silent about workplace harassment, 61 per cent said their employers did not coerce them to keep quiet, but a significant 35pc were told to remain silent by their colleagues and bosses.

To address the problem of identifying scams on job posting, supervised learning algorithm as classification techniques are considered initially. A classifier maps input variable to target classes by considering training data. Classifiers addressed in the paper for identifying fake job posts from the others are described

briefly. These classifiers-based predictions may be broadly categorized into -Single Classifier based Prediction and Ensemble Classifiers based Prediction.

1.2 Problem Statement

There are a lot of job advertisements on the internet, even on the reputed job advertising sites, which never seem fake. But after the selection, the so-called recruiters start asking for the money and the bank details. **[1]** Many of the candidates fall in their trap and lose a lot of money and the current job sometimes. So, it is better to identify whether a job advertisement posted on the site is real or fake. Identifying it manually is very difficult and almost impossible!

1.3 Objective

To avoid fraudulent posts for job in the internet, an automated online tool using machine learning based classification and techniques is proposed. Different machine learning classifiers and algorithms are used for checking fraudulent job posting in the web and the results of those machine learning classifiers and algorithms are compared for identifying the fraudulent job posting. It helps in detecting fraudulent job posts from an enormous number of posts on internet. Type of Machine Learning: Supervised Learning is used for classification and regression of fraudulent job posts predication on internet.

1.4 Project Scope

This project will develop and deliver a new online automated tool/website using Python. This new online automated tool/website will display a number of job posts which will be real not containing any fake post. The users will be able to apply online for job that contains real jobs which will save their money cost and time. This new online automated tool/website will contain a huge database of job posts as a record. This online automated tool/website will be developed to enable additional features to be added to it over a period of time and be easy to maintain.

1.5 Advantages of Proposed Solution

This proposed solution contains a lot of advantages that will be proved fruitful for online job seekers. Following are advantages of proposed solution:

- Easy to use

- Time saving
- Cost effective (Saving money)
- Contains a huge database of job posts

1.6 Relevance to Study Program

According to several studies, Review spam detection, [2] Email Spam detection, Fake news detection have drawn special attention in the domain of Online Fraud Detection. [3]

1.6.1. Review Spam Detection-

People 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 detects 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 uses dictionary or corpus to eliminate spam reviews.

1.6.2. Email Spam Detection-

Unwanted bulk mails, belong to the category of spam emails, often arrive to user mailbox. This may lead to unavoidable storage crisis as well as bandwidth consumption. To eradicate this problem, Gmail, Yahoo mail and Outlook service providers incorporate spam filters using Neural Networks. [4] 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.

1.6.3. 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, how a user is related to fake news. Features related to news content and social context are extracted and a machine learning model are imposed to recognize fake news.

1.7 Chapter Summary

Online Fake Job Posting Predication will guide job-seekers to get only legitimate offers from companies. For tackling Online Fake Job Posting, several machine learning algorithms are proposed as countermeasures in this Project.

Chapter 2

Existing System

2.41 Existing System

2.1.1 Literature review

The project Literature review elaborate that no existing system is still implemented or in running state like this project. Serval research studies are proposed but not exist system like this before. The literature review provides a significant insight into the domain of machine Learning. The studies conducted by people are taking wide variety of approaches There are many ongoing researches on these machine learning techniques. [5] Research studies seen in this section highlights the potential advantages and popularity of machine learning. Literature review enabled to grasp more knowledge regarding the different 19 algorithms. Most of the studies mentioned in this chapter have repeatedly discovered the key merits in relying on machine learning technology.

Table 2.1 Literature review

Sr.	Title	Author	Year	Remark
1	Fake Job Recruitment Detection Using Machine Learning Approach	Shawni Dutta, Prof.Samir Kumar	4- April 2020	Naive Bayes, K-nearest Neighbor, Decision Tree Classifier
2	Machine Learning and Job Posting Classification	Ibrahim M. Nasser, Amjad H. Alzaanin	9, September-2020	Multinomial Naive Bayes, Support Vector Machine, Decision Tree, Random Forest Classifier
3	Smart Fraud Detection Framework for Job Recruitments	Asad Mehboob, M. S. I. Malik	4-October 2020	NB, KNN, DT, SVM, Random forest (RF), XGBoost Classifier
4	Comparative study on various algorithms for detection of fake job postings	Dhanamma Jagli, Vishal Saroj Gupta	09 | SEP 2020	SVM, Logistic Regression, KNN, RF, DT Classifier

2.2 Chapter summary

In this chapter, project Literature review is explained, no existing system like it before.

Chapter 3

Requirement Engineering

3.1 Detailed description of Proposed System

To avoid fraudulent posts for job in the internet, an automated online tool (website) using machine learning based classification and techniques is proposed. [6] Different machine learning classifiers and algorithms are used for checking fraudulent job posting in the web and the results of those machine learning classifiers and algorithms are compared for identifying the fraudulent job posting. It helps in detecting fraudulent job posts from an enormous number of posts on internet. Type of Machine Learning: Supervised Learning is used for classification and regression of fraudulent job posts predication on internet.

3.2 Understanding the system

In the Proposed system, the dataset is used in the proposed methods for testing the overall performance of the approach. Before fitting this data to any classifier, some pre-processing techniques are applied to this dataset. Pre-processing techniques include missing values removal, stop-words elimination, irrelevant attribute elimination and extra space removal. This prepares the dataset to be transformed into categorical encoding in order to obtain a feature vector. This feature vectors are fitted to several classifiers.

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. [7] At first, the classifiers are trained using the 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 and F-measure score are used for evaluating the prediction for each of these classifiers. Finally, the classifier that has the best performance with respect to all the metrics is chosen as the best candidate model.

3.2.1 User involvement

User involvement in system development is becoming more salient due to the fact that this can lead to better designed products from the perspective of the customers. One way to secure this is for designers to work in conjunction with the users, enrolling them early on in the development process, when their contributions to the system design are thought to be fundamental. Furthermore, user involvement allows

for obtaining sufficient information about the initial system requirements, for assessing if a product meets the end users' requirements and needs, and for gathering data for the next version of the design.

3.2.2 Stakeholders

In this system the stakeholders are the Companies and End users who will use my system.

3.2.3 Domain

The Domain of this system is Data Science and Artificial Intelligence/ Python which is implemented in my system.

3.3 Requirements Engineering

Requirement's engineering is the process of eliciting stakeholder needs and desires and developing them into an agreed-upon set of detailed requirements that can serve as a basis for all subsequent development activities. The purpose of requirements engineering methodologies is to make the problem that is being stated clear and complete, and to ensure that the solution is correct, reasonable, and effective.

There are two types of requirements are as follows

- Functional Requirements
- Non-Functional Requirements

3.3.1. Functional Requirements (User/Candidate):

Table 3.1 Functional Requirements (User)

Requirement ID	Requirement Description	Must/Want
FR001	The candidate can be login if the candidate to get apply online for the job. It is a part of the website. Without proper login a user has might no access to apply any job. He/ She must input their user's name and password correctly.	Want

FR002	The candidate will be login after the candidate to view jobs are available on our website, by viewing the jobs of our website.	Want
FR003	After viewing the job, the candidate must be chosen the valid job post URL are to be provided for our website and then he/she can be to go to the next level.	Must
FR004	After viewing the candidate job are providing of our website and then he/she can be viewed the job status are available on our website and choose the After viewing the candidate job are provide of our website and then he/she can be viewed the job status are available on our website and choose the job in which he/she want to apply online job.	Must
FR005	After viewing the job and job category then he/she can be viewed the job category. Job category is that the website provides the candidates for which scale or designation.	Want
FR006	After viewing and choosing the category, job is provided and available on website and see prediction results that either job post is fake or real. he/she will logout successfully.	Must

3.3.2 Functional Requirement (Admin)

Table 3.2 Functional Requirement (Admin)

Requirement ID	Requirement Description	Must/Want
FR001	Admin must have a secure login and password to enter in the system. Admin can access all user's data here at any time. Admin can also access any activity and the job are provided on our website.	Want

FR002	If any user found a negative comment of our website or give wrong data. Then admin has enough rights to delete the user instantly.	Want
FR003	Admin can add the job category as per demand and need of the candidate and also provided and available to the candidate. All the new job of this category will be placed in this category.	Must
FR004	System should perform classification of job post and show the results on the dashboard so that the user can view the fake job post analysis and see the expected percentage of fake jobs post on internet per day. .	Want
FR005	Admin can edit the job category as they entered a wrong monogram or the miss-spelled word for the category that was creating ambiguity for the user.	Must
FR006	After all the function perform on the website of admin then admin log out.	Must

3.3.3. Non-Functional Requirements

Table 3.3 Non-Functional Requirements

Requirement ID	Requirement Description	Must/Want
NF001	The system is kept secure so that no one can access system admin site without login and password. Outside users are not allowed entering in our system. Only registered members are capable of working on the system.	Want

NF002	In case of Forget Password an option of password recovery is placed. User click on the forget password button. On clicking of the button, a password reset link is send to the user's email for verification. After verification user can view old password for their use.	Want
NF003	The data of the entire users are kept secure. No one can access the data of other members except admin.	Want
NF004	Response time is an important factor in the systems. Our system is quite simple to perform and also the response time of any request is reduced.	Want
NF005	The system is so interactive that communicates and allows for interaction with users. And by interaction, we don't just mean allowing users to “click” and “scroll”. Offering users with content that is amusing, collaborative, and engaging is the essential objective of an interactive system.	Want
NF006	Ease of use is a basic concept that describes how easily users can use a system. Design teams define specific metrics per project—e.g., “Users must be able to tap Find within 3 seconds of accessing the interface.”—and aim to optimize ease of use while offering maximum functionality and respecting business limitations.	Want

3.3.6. Gantt Chart

TASK NAME	TASK START DATE	TASK END DATE	START ON DAY*	DURATION* (WORK DAYS)	PERCENT COMPLETE
First Sample Project					
Frame the Problem	9/15	9/28	0	14	100%
Define Project objectives	9/28	10/11	13	14	80%
Dataset Gathering	10/11	10/24	26	14	60%
Data Preprocessing	10/24	11/7	39	15	40%
Feature engineering	11/7	11/19	53	13	20%
Second Sample Project					
Dataset Partitioning	11/19	12/1	65	13	100%
Prepare Functional requirements	12/1	12/14	77	14	80%
ML Model Selection	12/14	12/28	90	15	60%
ML Model Training	12/28	1/10	104	14	50%
Validate ML Model	1/10	1/22	117	13	40%
Third Sample Project					
Design Web FrontEnd	2/15	3/3	153	17	100%
Intergrate ML Model Into Web	3/4	3/21	170	18	80%
Make Classfication and Perdiction	3/22	4/9	188	19	60%
Fourth Sample Project					
System Testing	4/10	4/26	207	17	100%
Productionize System	4/27	5/13	224	17	80%
Launch System	5/14	5/30	241	17	60%
Monitor and Maintain	6/1	6/11	259	11	50%

Figure 3.1 Gantt Chart

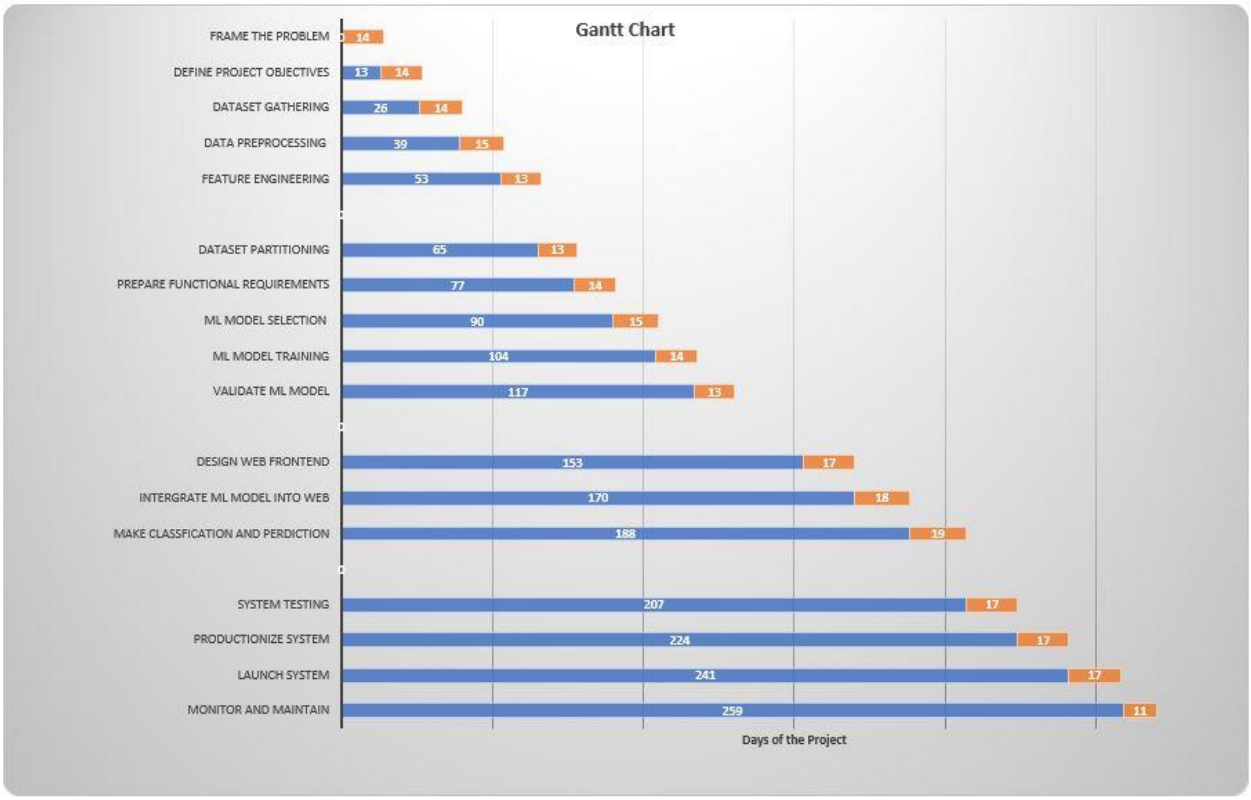


Figure 3.2 Gantt Chart

3.4 Hurdles in optimizing the current system

There are some major hurdles in optimizing the current system to achieve optimized performance that are as follows.

- Data preprocessing is the basic huddle of our current system because I have to done clean data free from errors.
- Model Selection is also a hurdle because a lot of existing models are proposed and I have to find best fit model among them
- Time bond Fake E job Posting prediction require time bond to show result to end user with in limited time.

3.5 Chapter Summary

This chapter summarizes available representative requirements engineering methodologies, mainly focusing on the principles. Requirements engineering approaches are processes that develop real-world problems into digital world solutions. Each approach has its specialized thinking about the real-world problem and follows a unique process to build the system specification as the solution.

Chapter 4

Design

4.1 Software Process Model

The Waterfall Model was the first Process Model to be introduced. It is also referred to as a linear-sequential life cycle model. It is very simple to understand and use. In a waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases.

The Waterfall model is the earliest SDLC approach that was used for software development.

The waterfall Model illustrates the software development process in a linear sequential flow. This means that any phase in the development process begins only if the previous phase is complete. In this waterfall model, the phases do not overlap.

4.2 Benefits of Model

The Benefits of waterfall development are that it allows for departmentalization and control. A schedule can be set with deadlines for each stage of development and a product can proceed through the development process model phases one by one.

Development moves from concept, through design, implementation, testing, installation, troubleshooting, and ends up at operation and maintenance. Each phase of development proceeds in strict order.

Some of the major advantages of the Waterfall Model are as follows −

- Simple and easy to understand and use
- Easy to manage due to the rigidity of the model. Each phase has specific deliverables and a review process.
- Phases are processed and completed one at a time.
- Works well for smaller projects where requirements are very well understood.
- Clearly defined stages.
- Well understood milestones.

- Easy to arrange tasks.
- Process and results are well documented.

4.3 Limitations of Model

The model implies that you should attempt to complete a given stage before moving on to the next stage Does not account for the fact that requirements constantly change.
It also means that customers cannot use anything until the entire system is complete.

- The model makes no allowances for prototyping.
- It implies that you can get the requirements right by simply writing them down and reviewing them.
- The model implies that once the product is finished, everything else is maintenance.

The waterfall model assumes that the requirements of a system can be frozen (i.e. based line) before the design begins. This is possible for systems designed to automate an existing manual system. But for absolutely new system, determining the requirements is difficult, as the user himself does not know the requirements. Therefore, having unchanging (or changing only a few) requirements is unrealistic for such project.

Freezing the requirements usually requires choosing the hardware (since it forms a part of the requirement specification). A large project might take a few years to complete. If the hardware is selected early, then due to the speed at which hardware technology is changing, it is quite likely that the final software will employ a hardware technology that is on the verge of becoming obsolete. This is clearly not desirable for such expensive software.

The waterfall model stipulates that the requirements should be completely specified before the rest of the development can proceed. In some situations, it might be desirable to first develop a part of the system completely, an then later enhance the system in phase. This is often done for software products that are developed not necessarily for a client (where the client plays an important role in requirement specification), but for general marketing, in which the requirements are likely to be determined largely by developers.

4.4. Design

Waterfall approach was first SDLC Model to be used widely in Software Engineering to ensure success of the project. In "The Waterfall" approach, the whole process of software development is divided into separate phases. In this Waterfall model, typically, the outcome of one phase acts as the input for the next phase sequentially.

The following illustration is a representation of the different phases of the Waterfall Model.

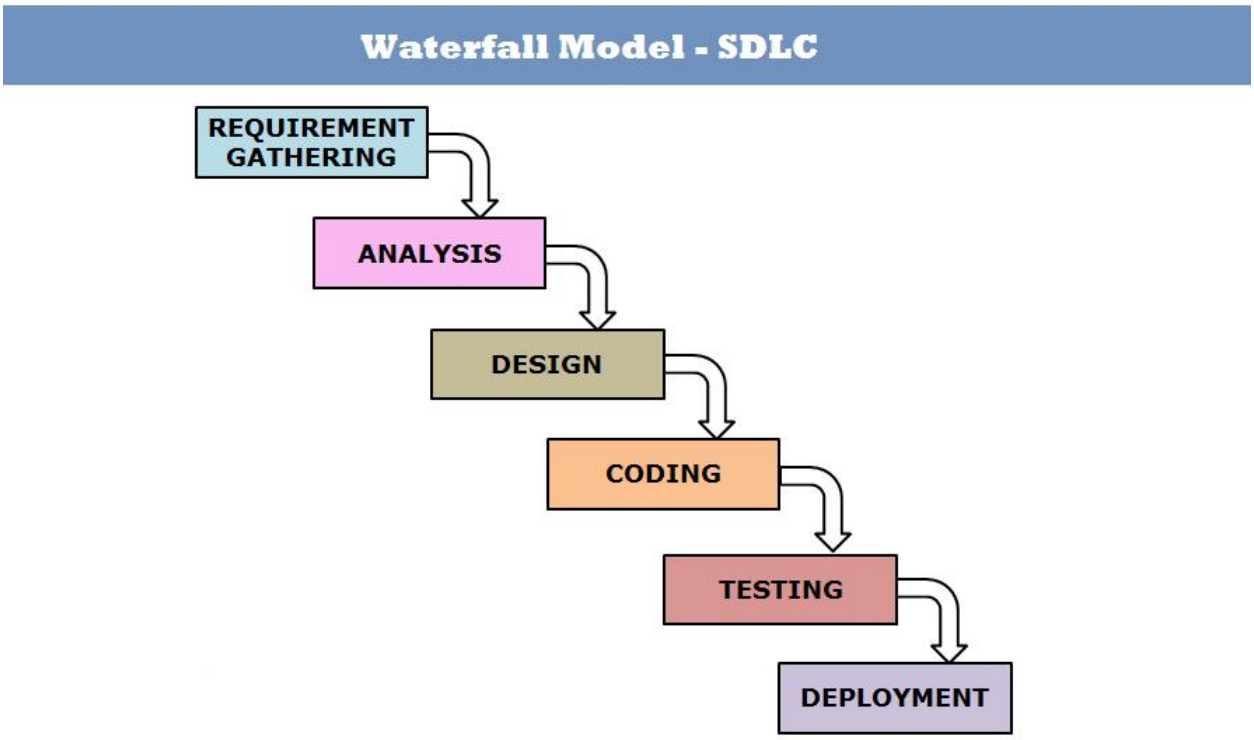


Figure 4.1 Waterfall Model

The sequential phases in Waterfall model are

- **Requirement Gathering and analysis** − All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document. [8]
- **System Design** − The requirement specifications from first phase are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and helps in defining the overall system architecture.
- **Implementation** − With inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality, which is referred to as Unit Testing.
- **Integration and Testing** − All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.
- **Deployment of system** − Once the functional and non-functional testing is done; the product is deployed in the customer environment or released into the market.
- **Maintenance** − There are some issues which come up in the client environment. To fix those issues, patches are released. Also, to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

All these phases are cascaded to each other in which progress is seen as flowing steadily downwards (like a waterfall) through the phases. The next phase is started only after the defined set of goals are achieved for previous phase and it is signed off, so the name "Waterfall Model". In this model, phases do not overlap.

4.5 Methodology Diagram

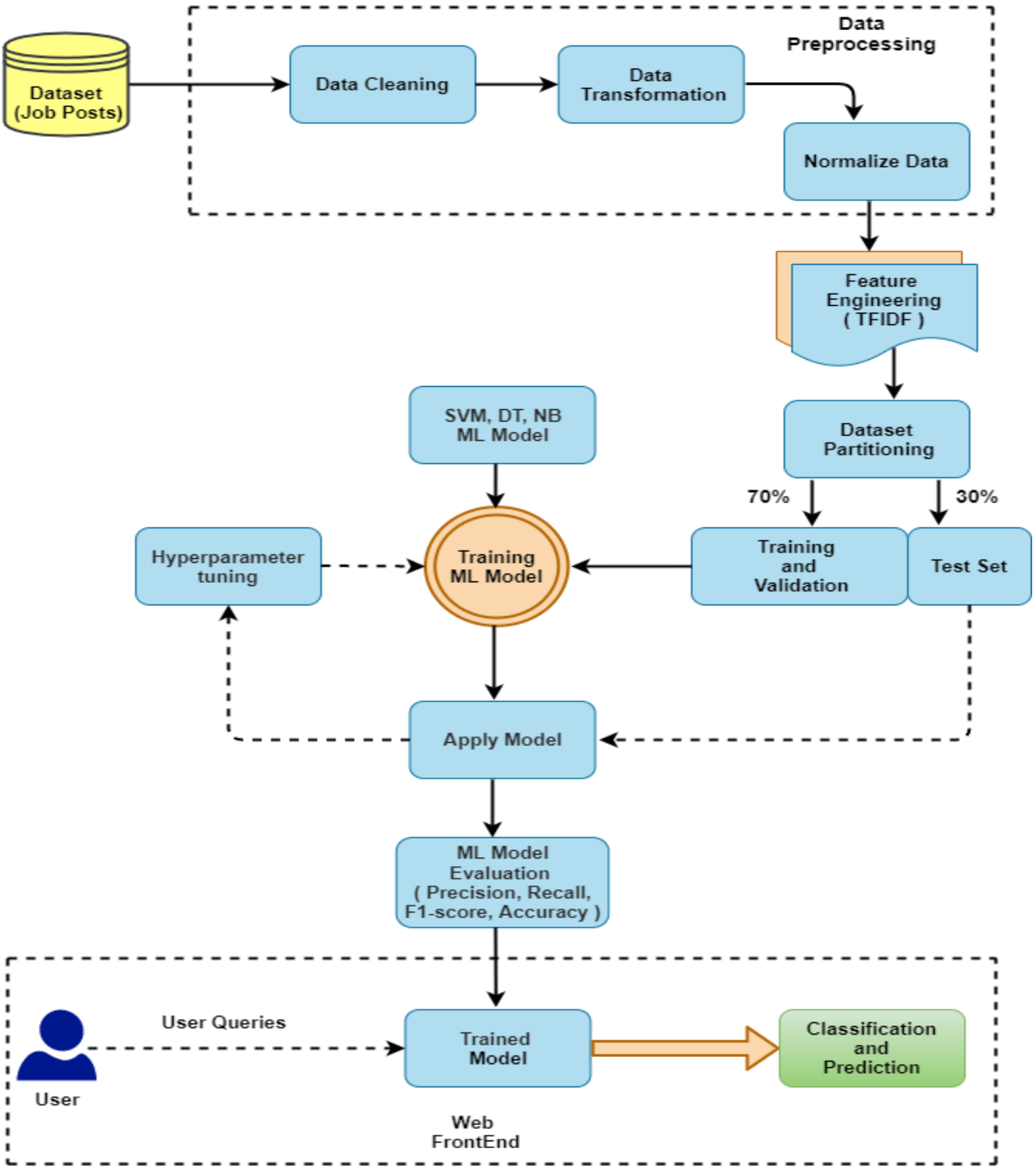


Figure 4.2 Methodology Diagram

4.6 System Flow Diagram

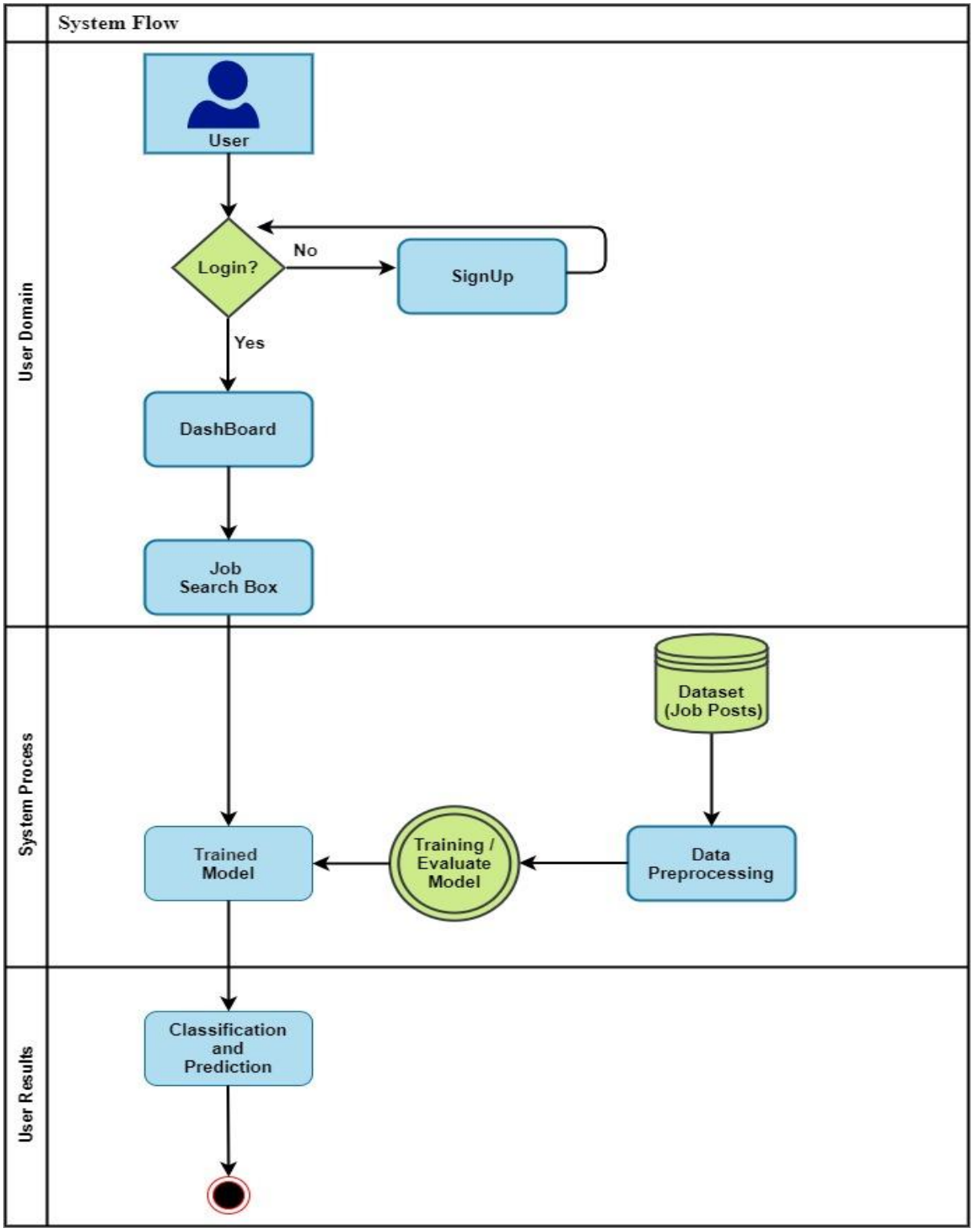


Figure 4.3 System Flow Diagram

4.7 UML Diagrams

UML is an acronym that stands for **Unified Modeling Language**. Simply put, UML is a modern approach to modeling and documenting software. In fact, it's one of the most popular business process modeling techniques.

It is based on **diagrammatic representations** of software components. As the old proverb says: "a picture is worth a thousand words". By using visual representations, we are able to better understand possible flaws or errors in software or business processes.

4.8 Collaboration Diagram

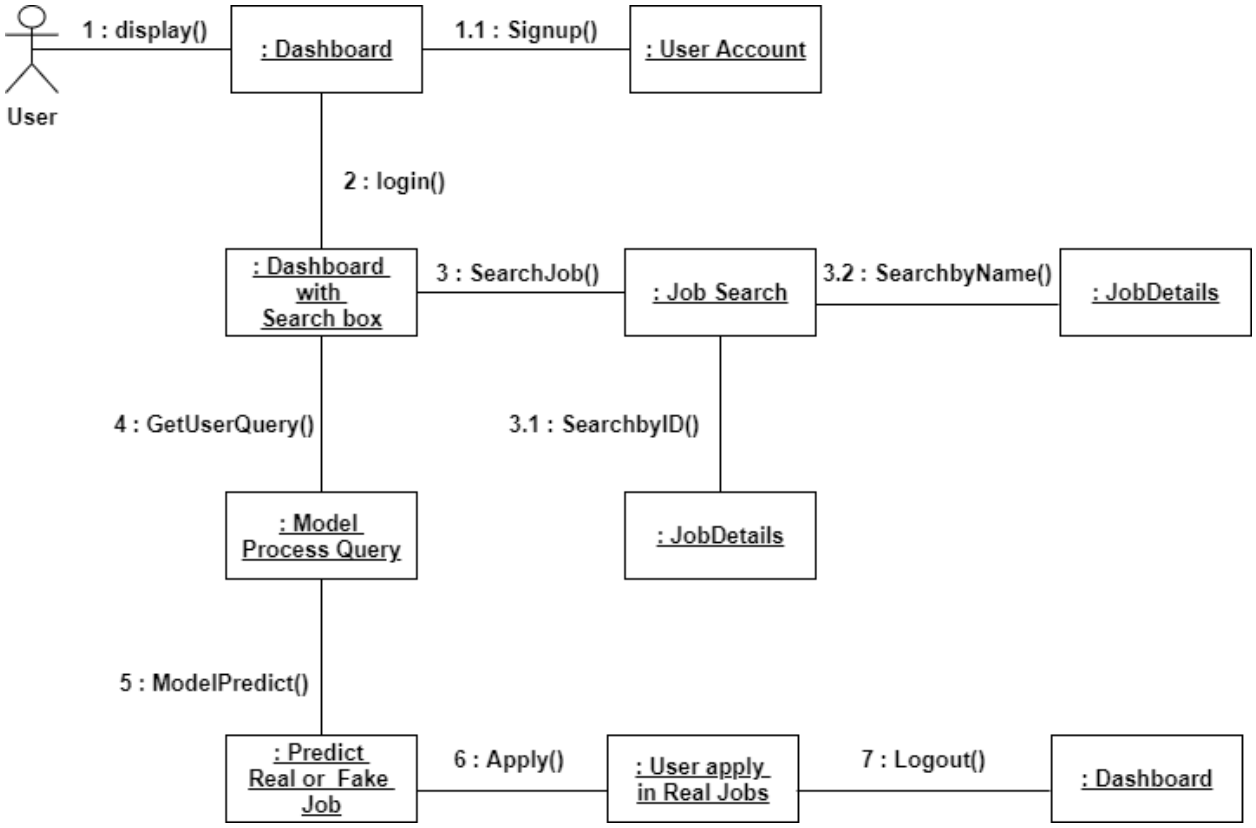


Figure 4.4 Collaboration Diagram

4.9 Activity Diagram

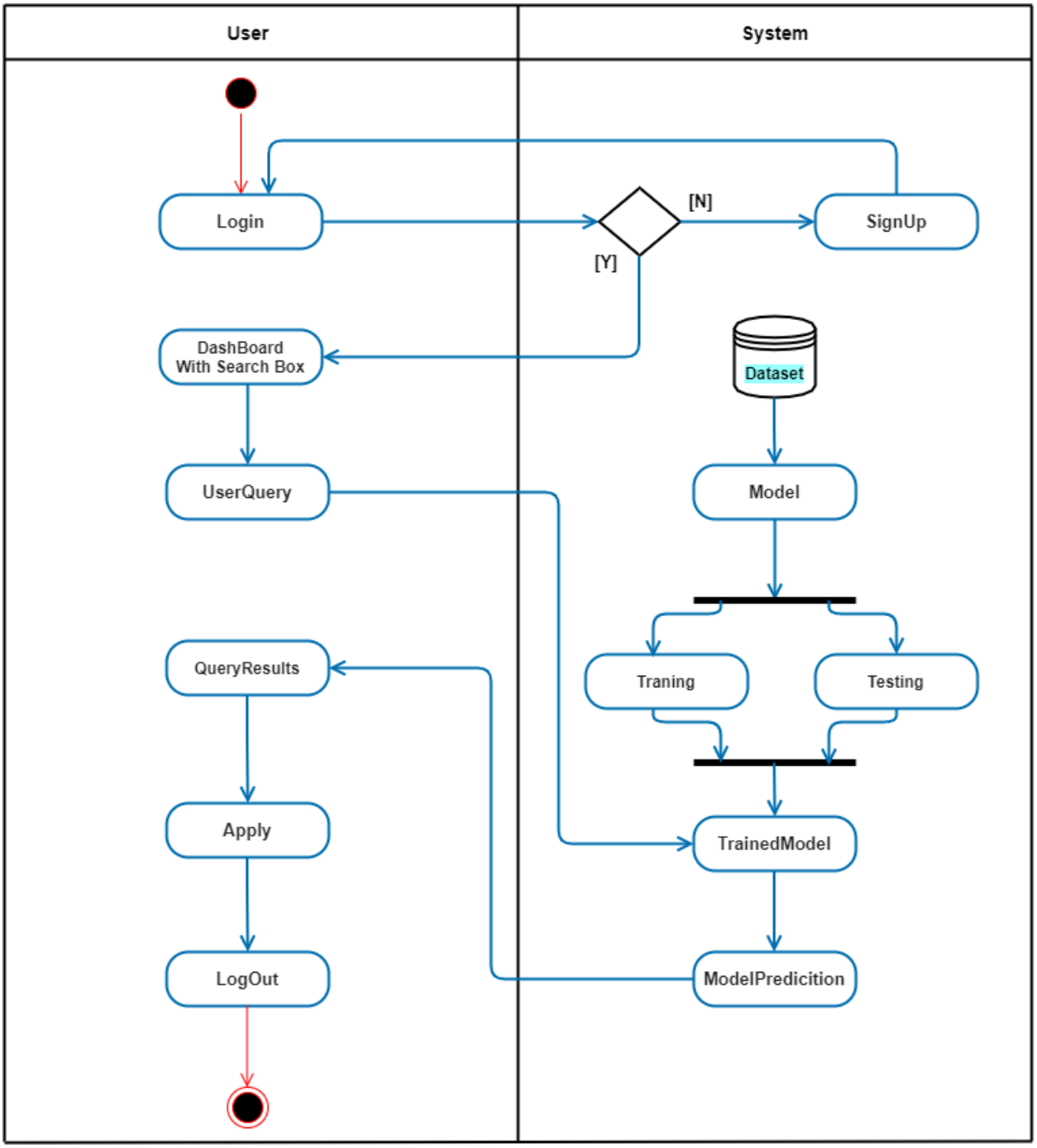


Figure 4.5 Activity Diagram

4.10 Use Case Diagram

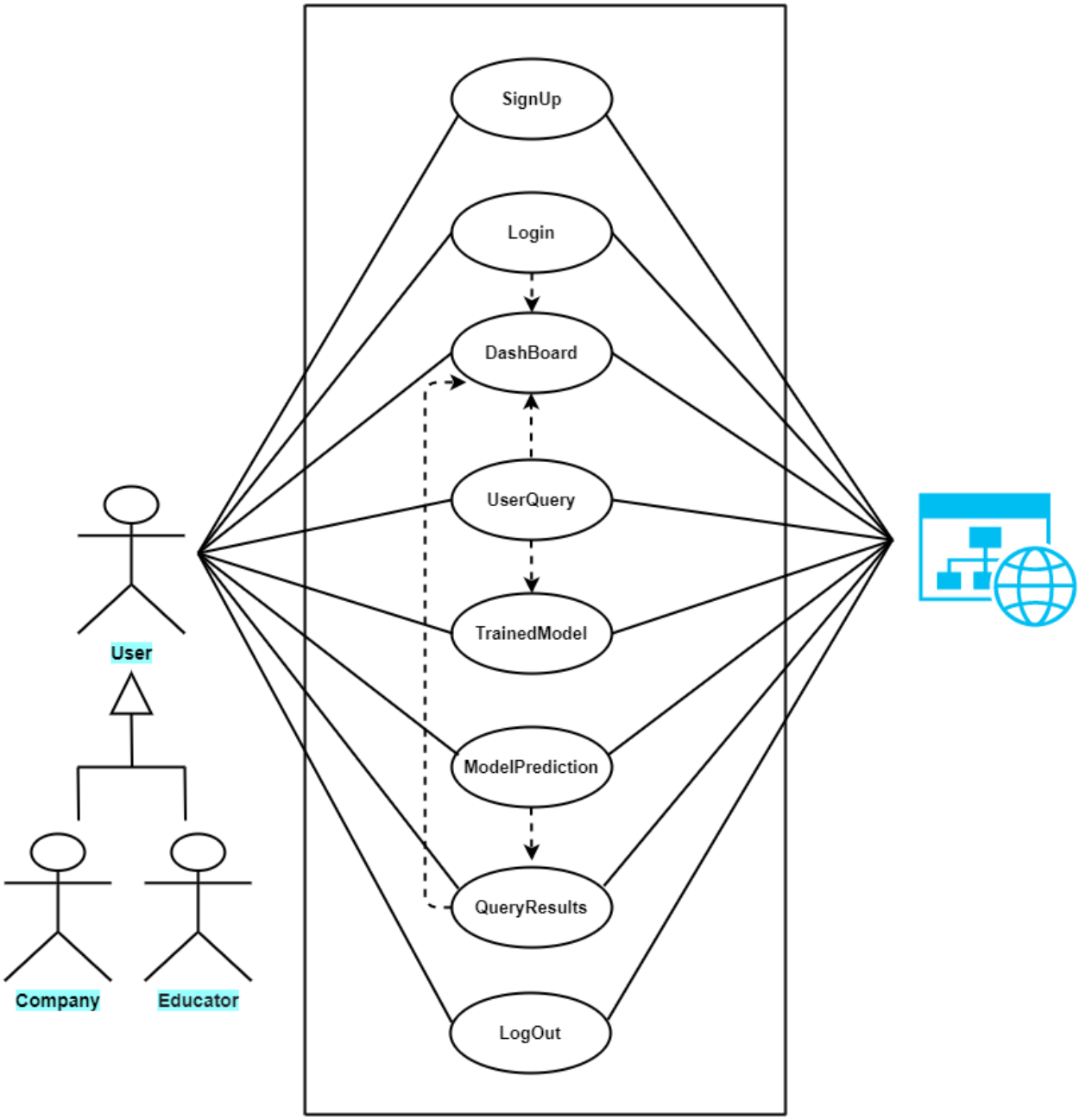


Figure 4.6 Use Case Diagram

4.11 Chapter Summary

UML is a common language for business analysts, software architects and developers used to describe, specify, design, and document existing or new business processes, structure and behavior of artifacts of software systems. UML can be applied to diverse application domains (e.g., banking, finance, internet, aerospace, healthcare, etc.) It can be used with all major object and component software development methods and for various implementation platforms. In this chapter all UML diagrams is completed and visualize here perfectly

Chapter 5

Dataset Introduction

5.1 Dataset description

The Employment Scam Aegean Dataset (EMSCAD) is a publicly available dataset containing 17,880 real-life job ads that aims at providing a clear picture of the Employment Scam problem to the research community and can act as a valuable testbed for scientists working on the field. [9]

EMSCAD records were manually annotated and classified into two categories. More specifically, the dataset contains 17,014 legitimate and 866 fraudulent job ads published between 2012 to 2014.
Emails, phones and URLs found in texts were masked via the pattern *#(EMAIL|PHONE|URL)_Keyed_SHA2#*.
This dataset contains 18K job descriptions out of which about 800 are fake. The data consists of both textual information and meta-information about the jobs. The dataset can be used to create classification models which can learn the job descriptions which are fraudulent.

5.2 Attributes

5.2.1 String

Name	Description
Title	The title of the job ad entry.
Location	Geographical location of the job ad.
Department	Corporate department (e.g., sales).
Salary range	Indicative salary range (e.g., $50,000-$60,000)

Name	Description

5.2.2 HTML fragment

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.

5.2.3 Binary

Telecommuting	True for telecommuting positions.
Company logo	True if company logo is present.
Questions	True if screening questions are present.
Fraudulent	Classification attribute.
In balanced	Selected for the balanced dataset

5.2.4 Nominal

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.

5.3 Dataset type

The data consists of both textual information and meta-information about the jobs.

5.4 Import the Dataset

```
In [2]: 1 data = pd.read_csv(r'fake_job_postings.csv')

In [4]: 1 data.shape
Out[4]: (17880, 18)

In [3]: 1 data.columns
Out[3]: Index(['job_id', 'title', 'location', 'department', 'salary_range',
               'company_profile', 'description', 'requirements', 'benefits',
               'telecommuting', 'has_company_logo', 'has_questions', 'employment_type',
               'required_experience', 'required_education', 'industry', 'function',
               'fraudulent'],
              dtype='object')
```

5.5 Dataset Head

```
In [87]: 1 data.head()
```

Out[87]:

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

5.6 Check out the Missing Values

The concept of missing values is important to understand in order to successfully manage data. If the missing values are not handled properly by the researcher, then he/she may end up drawing an inaccurate inference about the data. [10] Due to improper handling, the result obtained by the researcher will differ from ones where the missing values are present.

```
In [3]: 1 data.interpolate(inplace=True)
        2 data.isnull().sum()
```

```
Out[3]: job_id                   0
        title                    0
        location               346
        department           11547
        salary_range         15012
        company_profile       3308
        description              1
        requirements          2695
        benefits              7210
        telecommuting            0
        has_company_logo         0
        has_questions            0
        employment_type       3471
        required_experience   7050
        required_education    8105
        industry              4903
        function              6455
        fraudulent               0
        dtype: int64
```

Filling null values with black space

```
In [5]:  1  data.fillna(' ', inplace=True)

In [91]: 1  data.head()
```

5.7 Correlation Diagram

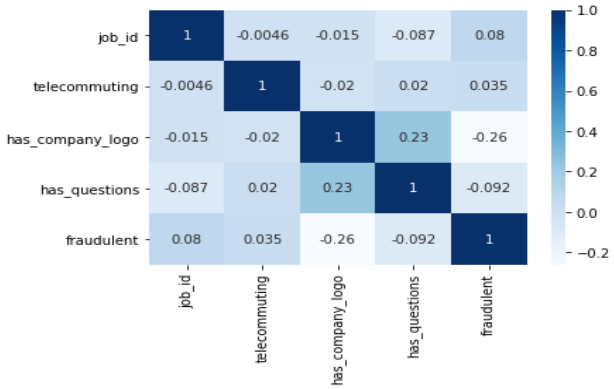


Figure 5.1 Correlation Diagram

5.8 Benefits

The dataset is very valuable as it can be used to answer the following questions:

1. Create a classification model that uses text data features and meta-features and predict which job description are fraudulent or real.
2. Identify key traits/features (words, entities, phrases) of job descriptions which are fraudulent in nature.
3. Run a contextual embedding model to identify the most similar job descriptions.
4. Perform Exploratory Data Analysis on the dataset to identify interesting insights from this dataset.

5.9 Constraints

1. The provided data to Model behind website must be a job post.
2. The job Post webpage URL must be a valid URL for predictions.

5.10 Chapter Summary

In this chapter, dataset is discussed. Detailed description of dataset and its features is provided here. This phase show that the dataset is complete and ready to train and test the Machine learning model and then to implement.

Chapter 6

Development

Development is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.

6.1 Data Pre-processing

Data pre-processing is a data mining technique that involves transforming raw data into an understandable format. Real-world data is often incomplete, inconsistent, and/or lacking in certain behaviours or trends, and is likely to contain many errors. Data pre-processing is a proven method of resolving such issues.

6.2 Why use Data Pre-processing?

In the real-world data are generally incomplete: lacking attribute values, lacking certain attributes of interest, or containing only aggregate data. Noisy: containing errors or outliers. Inconsistent: containing discrepancies in codes or names.

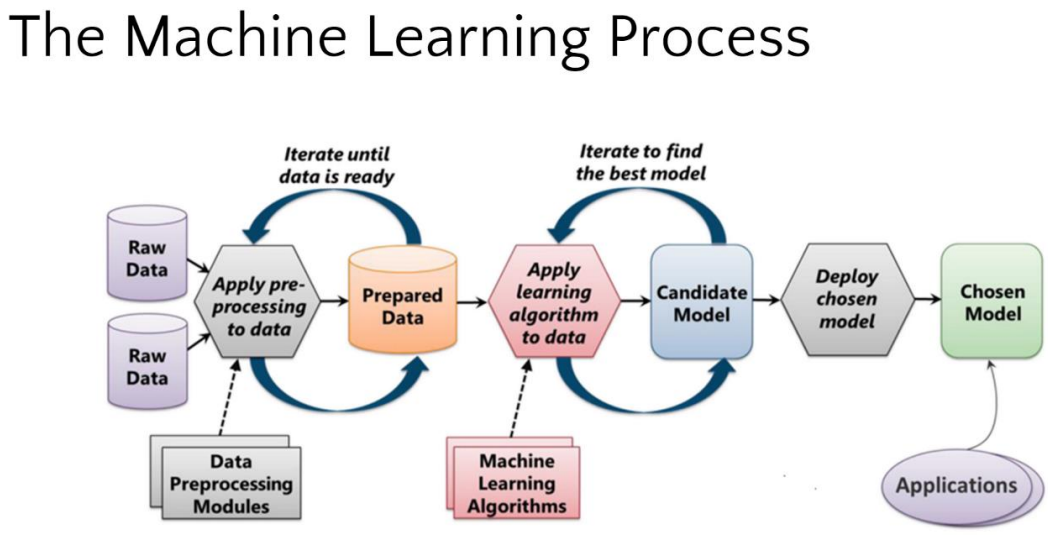


Figure 6.1 Machine learning process

6.3 Dataset Info

By using Pandas, we import our data-set and the file I used here is .csv file However, to access and to use fastly we use CSV files because of their light weights. After importing the dataset, you can see we use head function (This function returns the first n rows for the object based on position. It is useful for quickly testing if your object has the right type of data in it. By default, it returns 5 rows.)

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 17880 entries, 0 to 17879
Data columns (total 18 columns):
 #   Column               Non-Null Count  Dtype 
---  ------               --------------  ----- 
 0   job_id               17880 non-null  int64 
 1   title                17880 non-null  object
 2   location             17534 non-null  object
 3   department           6333 non-null   object
 4   salary_range         2868 non-null   object
 5   company_profile      14572 non-null  object
 6   description          17879 non-null  object
 7   requirements         15185 non-null  object
 8   benefits             10670 non-null  object
 9   telecommuting        17880 non-null  int64 
 10  has_company_logo     17880 non-null  int64 
 11  has_questions        17880 non-null  int64 
 12  employment_type      14409 non-null  object
 13  required_experience  10830 non-null  object
 14  required_education   9775 non-null   object
 15  industry             12977 non-null  object
 16  function             11425 non-null  object
 17  fraudulent           17880 non-null  int64 
dtypes: int64(5), object(13)
memory usage: 2.5+ MB
```

Figure 6.2 dataset info

6.4 Machine Learning Process Steps in Data Pre-processing

6.4.1 Import the Libraries

In [1]:

```
import re
import string
import numpy as np
import pandas as pd
import random
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.feature_extraction.text import TfidfVectorizer, CountVectorizer
from sklearn.model_selection import train_test_split
from sklearn.pipeline import Pipeline
from sklearn.base import TransformerMixin
from sklearn.metrics import accuracy_score, plot_confusion_matrix, classification_report, confusion_matrix
from wordcloud import WordCloud
import spacy
from spacy.lang.en.stop_words import STOP_WORDS
from spacy.lang.en import English
from sklearn.svm import SVC
import warnings
warnings.filterwarnings('ignore')
```

This is how we import libraries in Python using import keyword and this is the most popular libraries which any Data Scientist used. (I used- Jupyter Notebook)

6.4.2 Splitting the data-set into Training and Test Set

In any Machine Learning model is that we're going to split data-set into two separate sets

- Training Set
- Test Set

6.5 Why we need splitting?

Well, here it's your algorithm model that is going to learn from your data to make predictions. Generally, we split the data-set into 70:30 ratio or 80:20 what does it mean, 70 percent data take in train and 30 percent data take in test. However, this Splitting can be varying according to the data-set shape and size.

```
In [17]:
1 X_train, X_test, y_train, y_test = train_test_split(data.text, data.fraudulent, test_size=0.3)

In [18]:
1 # Train-test shape
2 print(X_train.shape)
3 print(y_train.shape)
4 print(X_test.shape)
5 print(y_test.shape)

(12516,)
(12516,)
(5364,)
(5364,)
```

6.6 Implementation of Classifier

The target of this system 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.

6.7 Support Vector Machine (SVM)

6.7.1 What is 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 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). [2]

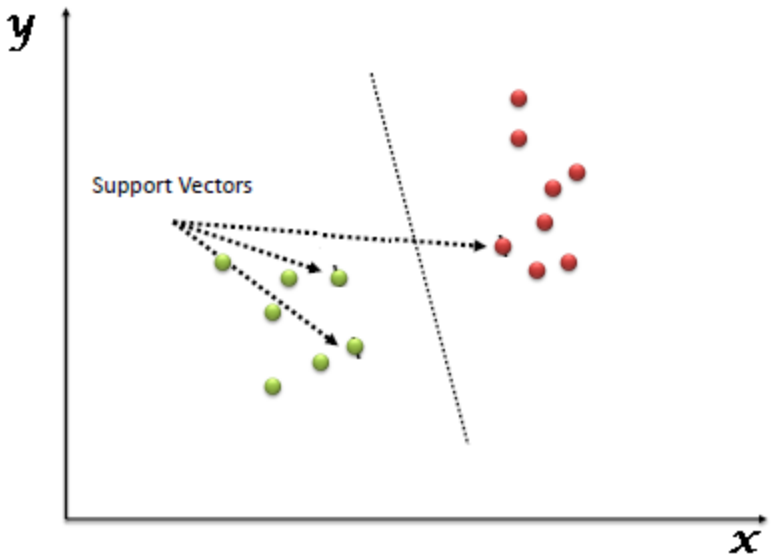


Figure 6.3 SVM

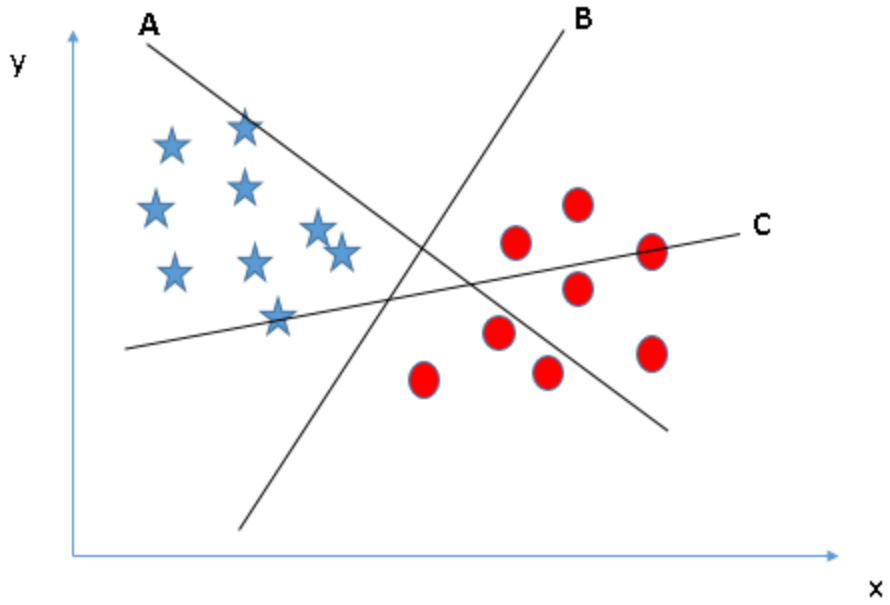
Support Vectors are simply the co-ordinates of individual observation. The SVM classifier is a frontier which best segregates the two classes (hyper-plane/ line).

6.7.2 How does it work?

Above, we got accustomed to the process of segregating the two classes with a hyper-plane. Now the burning question is “How can we identify the right hyper-plane?”.

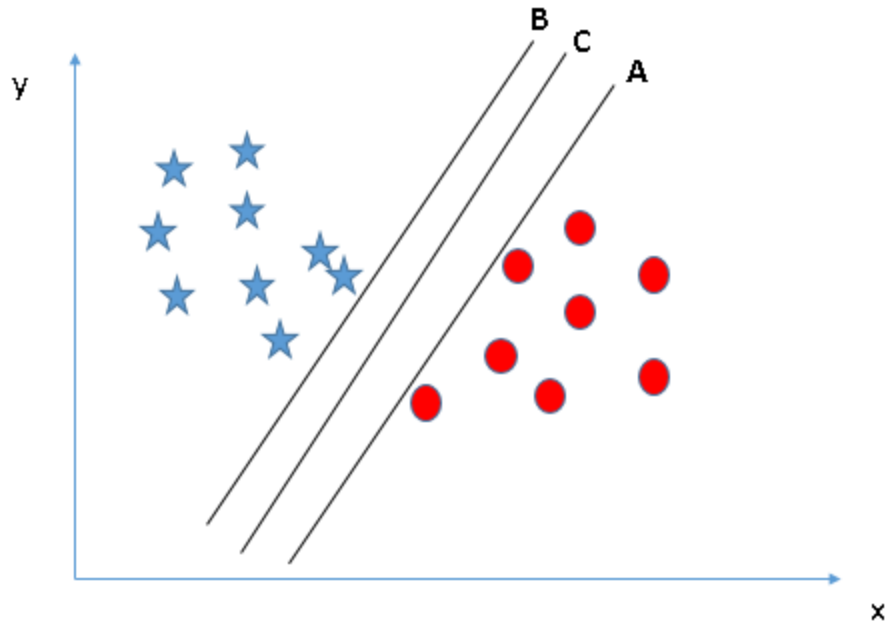
Let’s understand:

- **Identify the right hyper-plane (Scenario-1):** Here, we have three hyper-planes (A, B and C). Now, identify the right hyper-plane to classify star and circle.

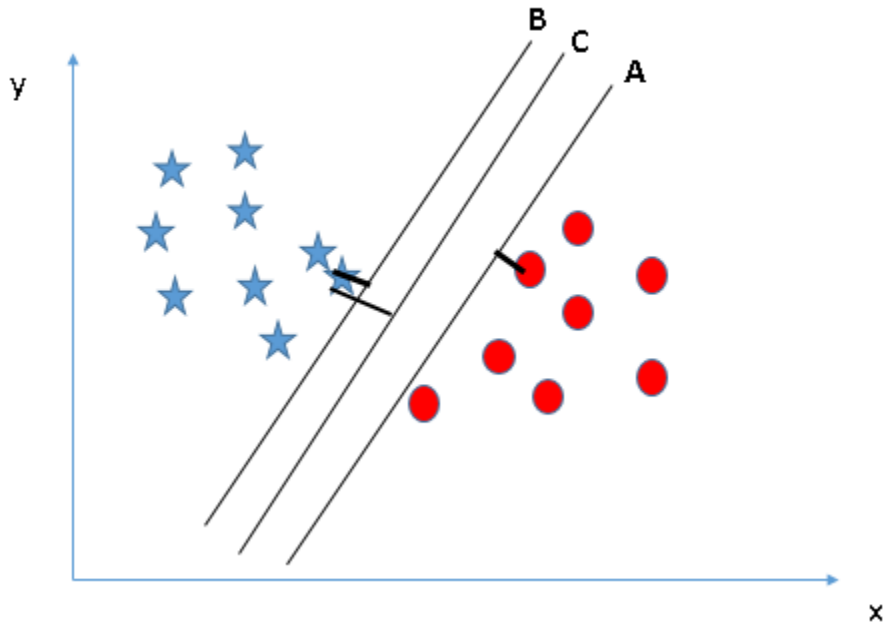


You need to remember a thumb rule to identify the right hyper-plane: "Select the hyper-plane which segregates the two classes better". In this scenario, hyper-plane "B" has excellently performed this job.

- **Identify the right hyper-plane (Scenario-2):** Here, we have three hyper-planes (A, B and C) and all are segregating the classes well. Now, how can we identify the right hyper-plane?



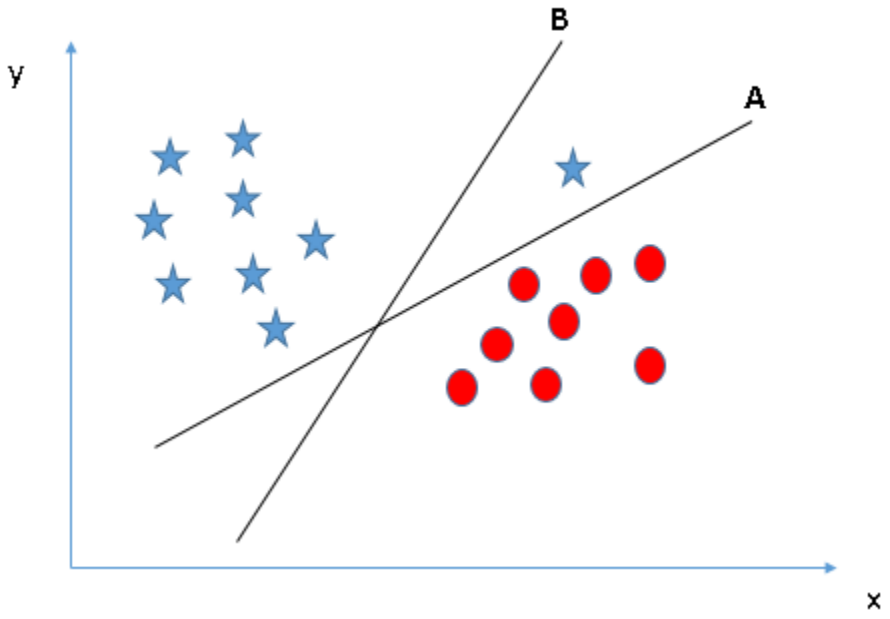
Here, maximizing the distances between nearest data point (either class) and hyper-plane will help us to decide the right hyper-plane. This distance is called as **Margin**. Let's look at the below snapshot:



Above, you can see that the margin for hyper-plane C is high as compared to both A and B. Hence, we name the right hyper-plane as C. Another lightning reason for selecting the hyper-

plane with higher margin is robustness. If we select a hyper-plane having low margin then there is high chance of miss-classification.

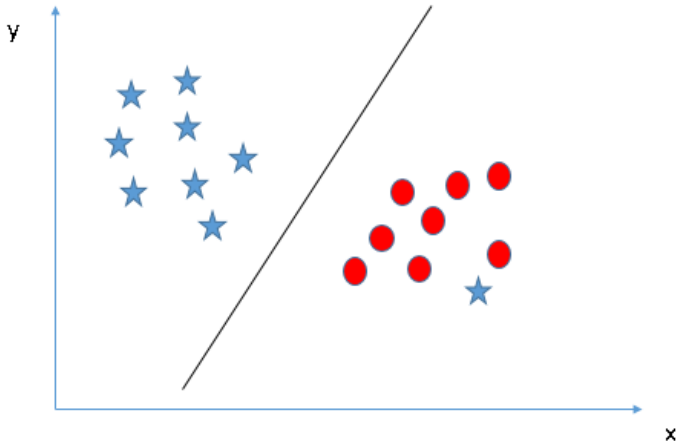
- **Identify the right hyper-plane (Scenario-3):** Hint: Use the rules as discussed in previous section to identify the right hyper-plane



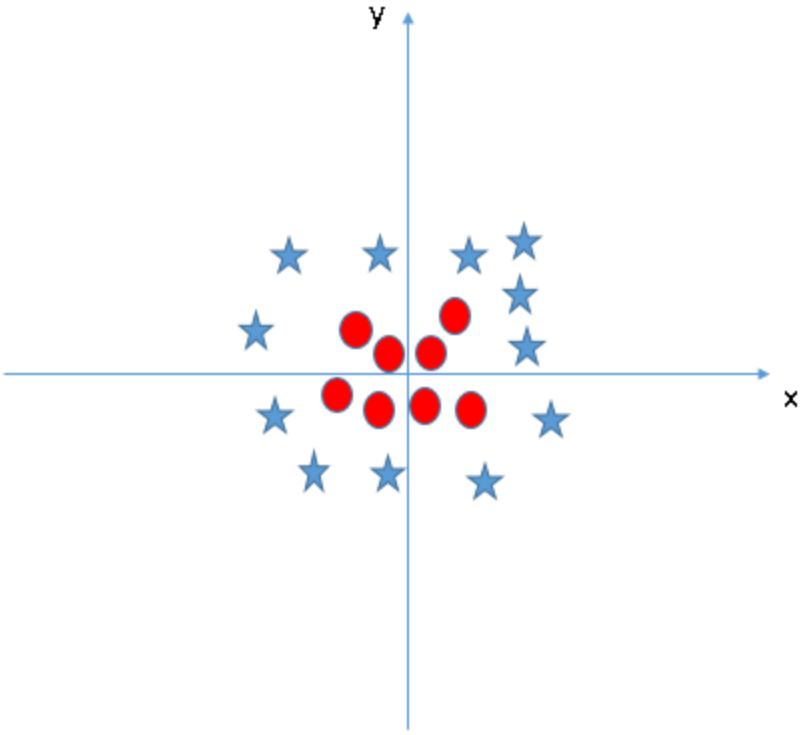
Some of you may have selected the hyper-plane **B** as it has higher margin compared to **A.** But here is the catch, SVM selects the hyper-plane which classifies the classes accurately prior to maximizing margin. Here, hyper-plane B has a classification error and A has classified all correctly. Therefore, the right hyper-plane is **A.**

- **Can we classify two classes (Scenario-4)?** Below, I am unable to segregate the two classes using a straight line, as one of the stars lies in the territory of other(circle) class as an outlier.

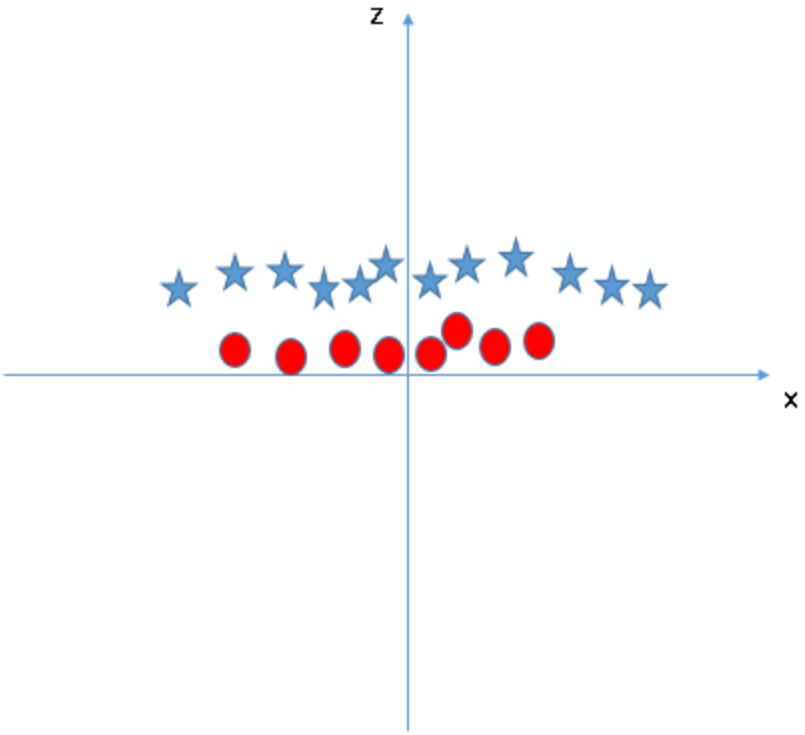
As I have already mentioned, one star at another end is like an outlier for star class. The SVM algorithm has a feature to ignore outliers and find the hyper-plane that has the maximum margin. Hence, we can say, SVM classification is robust to outliers.



- **Find the hyper-plane to segregate to classes (Scenario-5):** In the scenario below, we can't have linear hyper-plane between the two classes, so how does SVM classify these two classes? Till now, we have only looked at the linear hyper-plane.



SVM can solve this problem. Easily! It solves this problem by introducing additional feature. Here, we will add a new feature z=x^2+y^2. Now, let's plot the data points on axis x and z:

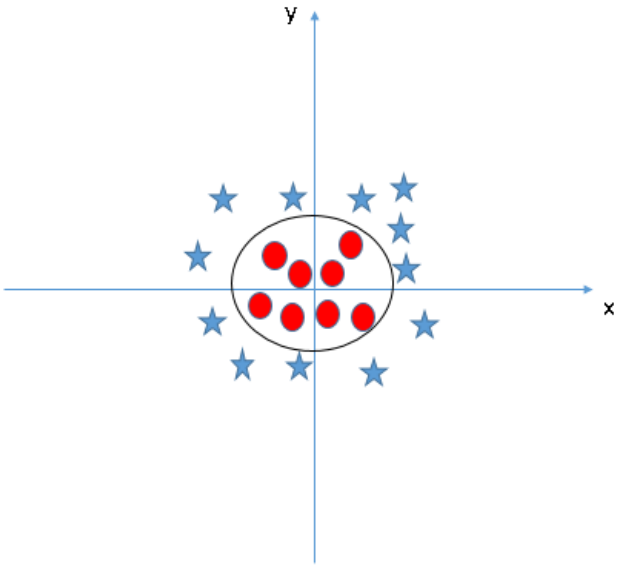


In above plot, points to consider are:

- All values for z would be positive always because z is the squared sum of both x and y
- In the original plot, red circles appear close to the origin of x and y axes, leading to lower value of z and star relatively away from the origin result to higher value of z.

In the SVM classifier, it is easy to have a linear hyper-plane between these two classes. But another burning question which arises is, should we need to add this feature manually to have a hyper-plane. No, the SVM algorithm has a technique called the kernel trick. The SVM kernel is a function that takes low dimensional input space and transforms it to a higher dimensional space i.e., it converts not separable problem to separable problem. It is mostly useful in non-linear separation problem. Simply put, it does some extremely complex data transformations, then finds out the process to separate the data based on the labels or outputs you've defined.

When we look at the hyper-plane in original input space it looks like a circle:



6.8 Human Computer Interface

human-computer interface (HCI) The means of communication between a human user and a computer system, referring in particular to the use of input/output devices with supporting software. They have to be configured in a way that will facilitate an efficient and desirable interaction between a person and the computer.

6.9 Interface Screenshots

6.9.1 Home

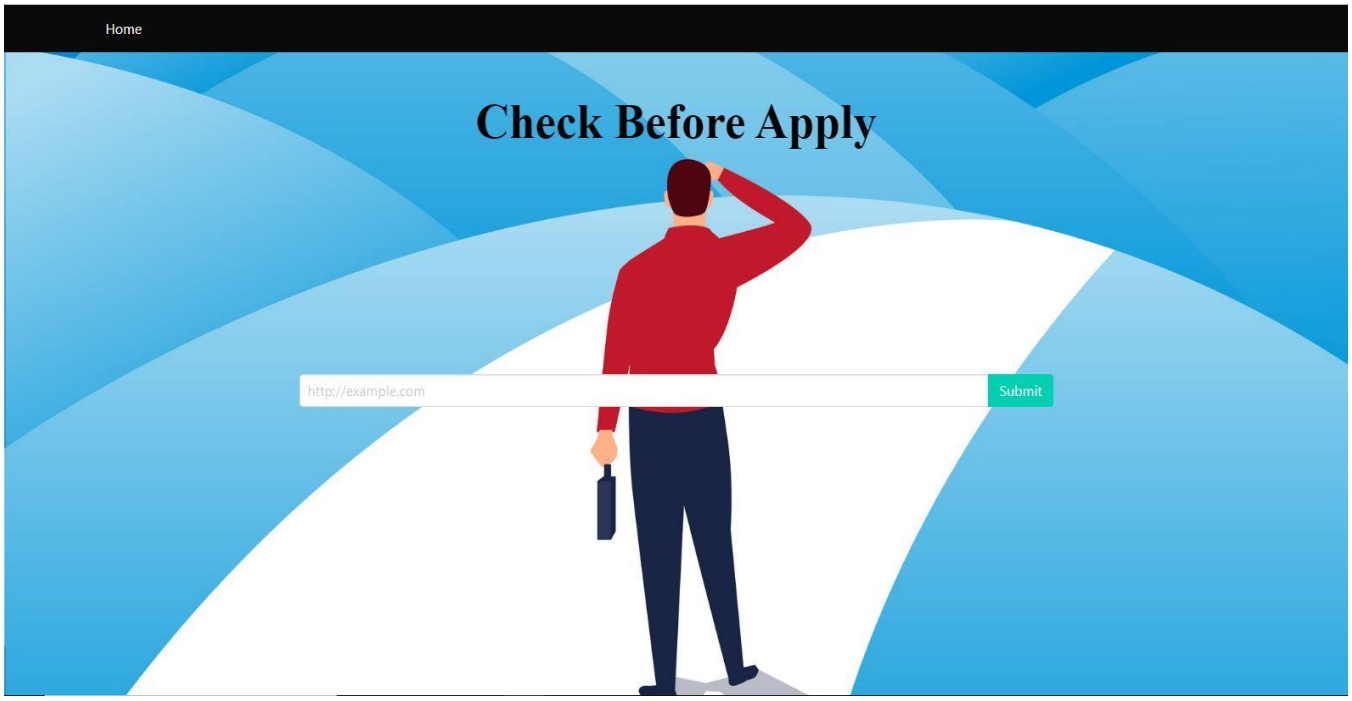
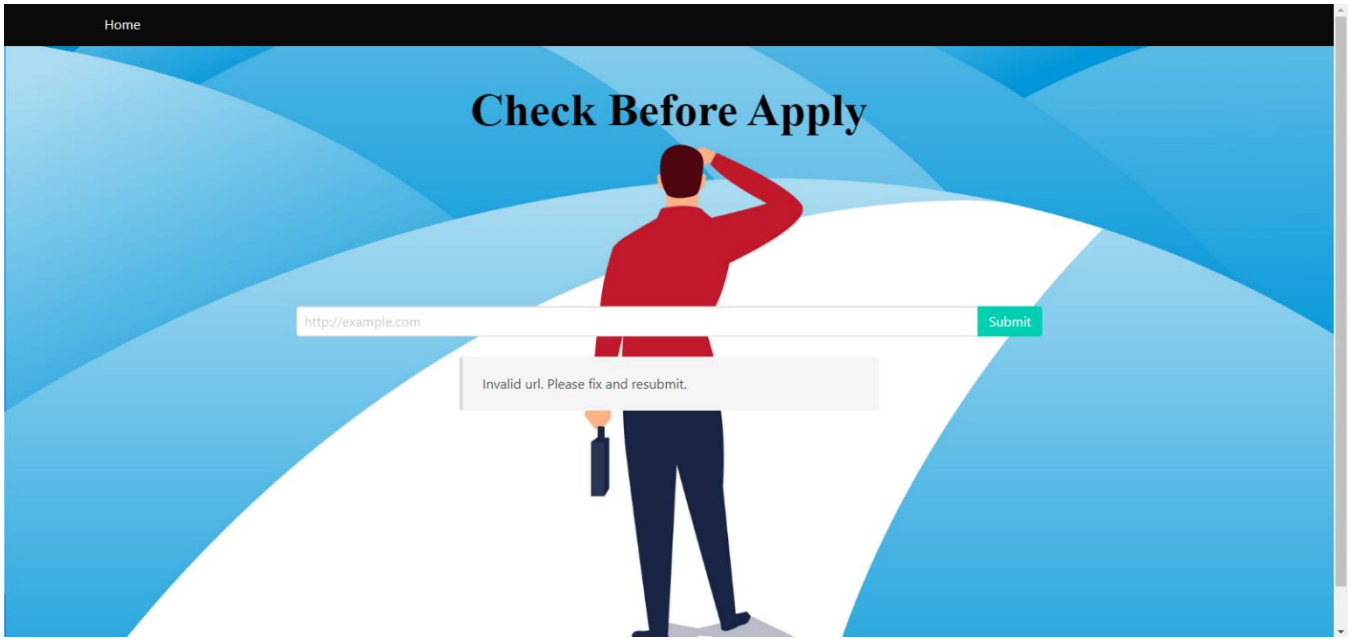


Figure 6.4 home page



6.9.2 Webpage URL Search Box

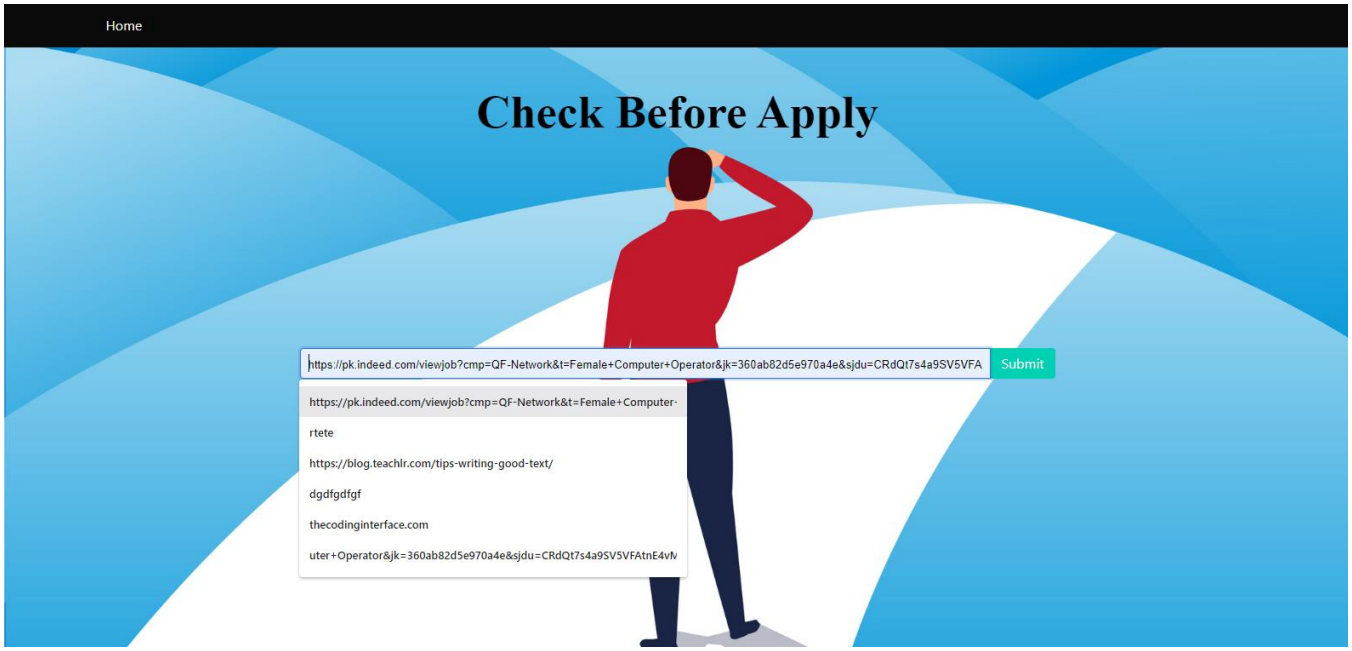
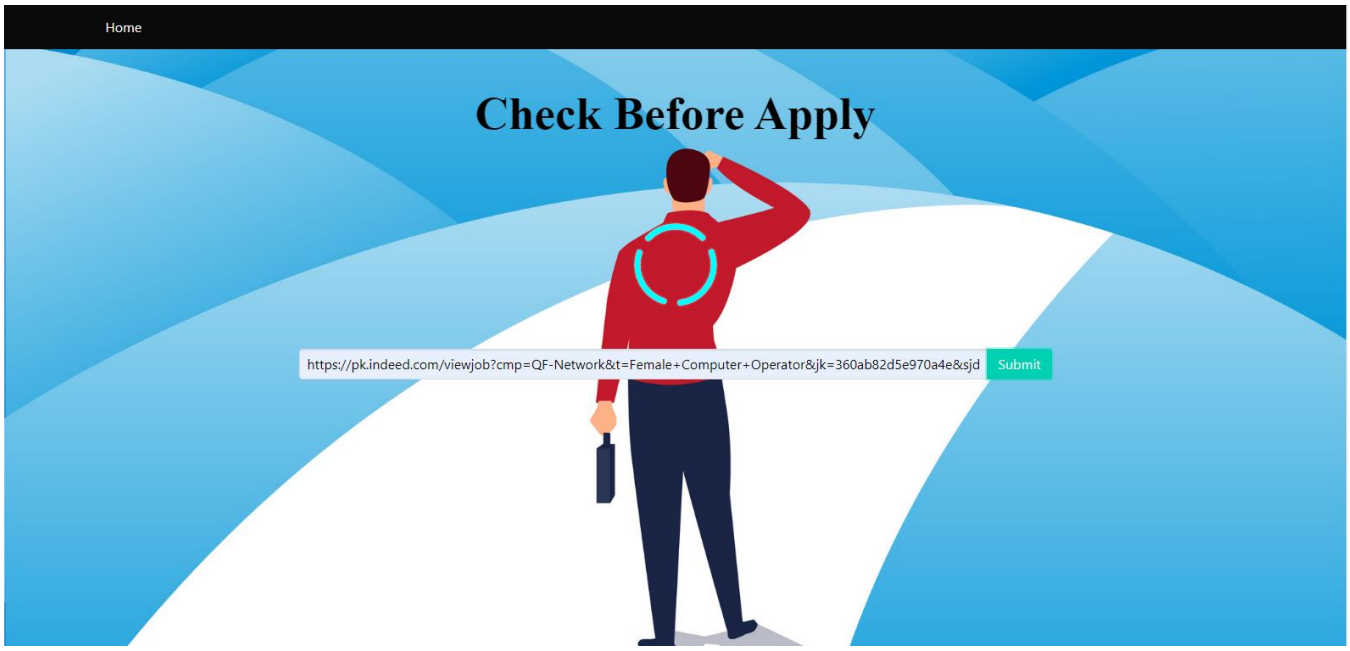


Figure 6.5 search page

6.9.3 Processing URL



6.9.4 Website Prediction Results

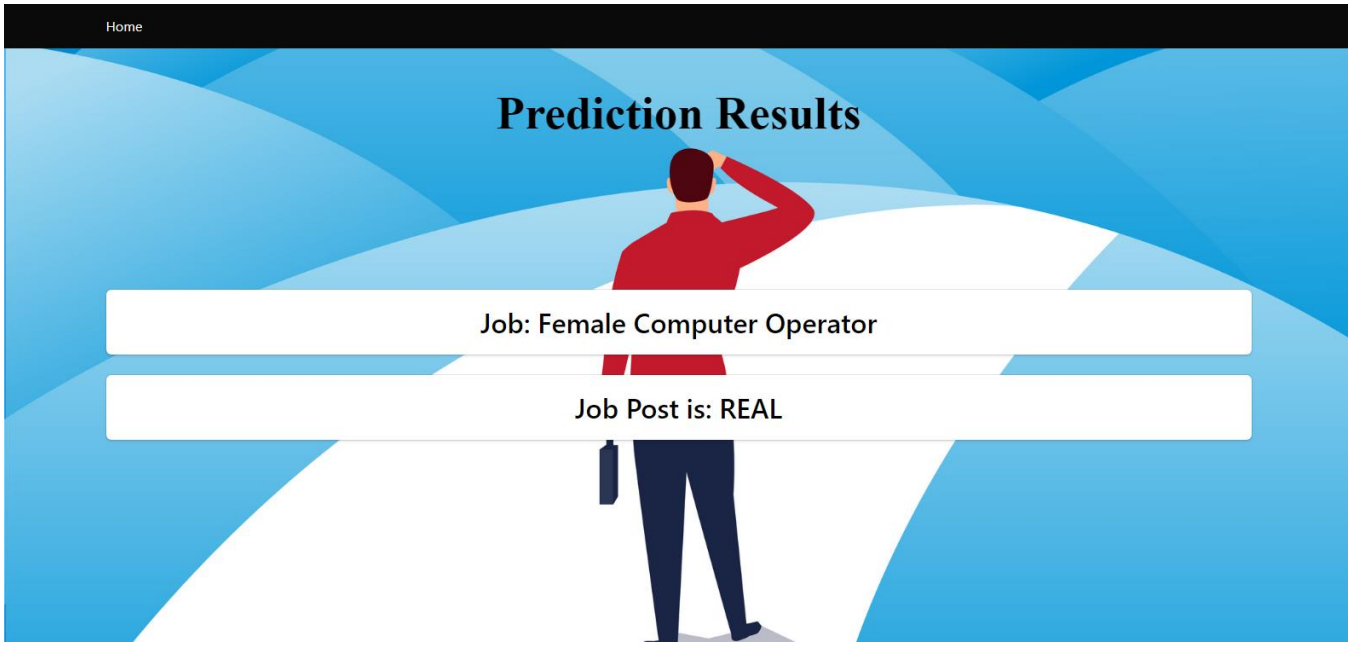


Figure 6.6 Results page

6.10 Chapter Summary

In this chapter, development phase is discussed. Development of web pages describe. And user interface screenshots from website also show to clear project view. This phase show that the project is complete and ready to test and then to implement.

Chapter 7

Testing

7.1 Testing

The Project is designed, implemented and tested incrementally (a little more is added each time) until the product is finished. It involves both development and maintenance. The product is defined as finished when it satisfies all of its requirements. First of all, I pre-processed the dataset and then build models on this dataset. It includes model training and testing. I develop the front end (user interface) for website by using HTML, CSS and Flask Framework. [5] After completing that phase, I tested that component either that is working or not, suits the environment and is appealing to user interaction or not. Second phase was creating Model connectivity. After creating that element, I tested the dataset is working according to the queries and properly being manipulated. After completing that task, I tested the last component. All these phases were created time to time and module was added incrementally.

7.2 Testing of Computer Program

7.2.1 System Test

System testing is also known as end-to-end testing. It tests a completely integrated system to verify that the system meets its requirements. [7] System testing falls within the scope of black-box testing. For example, a system test might involve testing a logon interface, then creating and editing an entry, plus sending email or printing reports, followed by summary processing or deletion of entries, then logoff.

7.2.2 Unit Test

In this type of testing, smallest testable parts of the system, units are individually tested and independently examined for correct functionality. This testing involves the both positive and negative testing. This is important so as to make sure that the system functions properly when used both correctly and incorrectly. In this case, the forms in android studio and Dreamweaver as well as the tables for the dataset will be tested individually to ensure that they are compatible. This also applies to the operating system and the software application. [9]

7.4 Test Case

Table 7.1 Test Case

Test Case ID	01
Application Name	Fake E Job Posting Prediction
Purpose	To describe the application form in which customer Provide job webpage URL.
Environment	Python, Flask
Pre-Requisite:	User will be the customer
Strategy:	User will perform the following operation a. Provide Job Post webpage URL. b. Show Results.
Expected Result	User should enter to the main form with their relevant privileges.
Observations	User have successfully entered the system having defined by admin
Test Case ID	02
Application Name	Fake E Job Posting Prediction
Purpose	To describe the application form in which customer Check the job post is real or fake
Environment	Python, Flask
Pre-Requisite:	User will be the customer
Strategy:	User will perform the following operation a. Submit the job post URL. b. check prediction results.
Expected Result	User should see their results successfully, if any incorrect information the error is occurred.
Observations	User have successfully entered the system having defined by admin

7.5 Future Work

At the end we conclude that our application will be the best application in the We have worked on those issues, solved them and implemented those amendments in our project. As well as further work is concern, we will definitely work on that to meet the latest technology and satisfy our candidate. In Future, we cover more jobs era and also add more categories of jobs in our application. That further amendment will be made according to the requirement of the future. Like adding Scholarships posts.

7.6. Chapter Summary

In this chapter, I conclude all work of project tested after develop. In different ways it is tested to check that there is no error. System works properly or not. Test cases also explain. Future work also discuss here.

Chapter 8

Results and Evaluation

While data preparation and training a machine learning model is a key step in the machine learning pipeline, it's equally important to measure the performance of this trained model. How well the model generalizes on the unseen data is what defines adaptive vs non-adaptive machine learning models.

By using different metrics for performance evaluation, we should be in a position to improve the overall predictive power of our model before we roll it out for production on unseen data.

Without doing a proper evaluation of the ML model using different metrics, and depending only on accuracy, it can lead to a problem when the respective model is deployed on unseen data and can result in poor predictions.

8.1 Plotting of dataset features

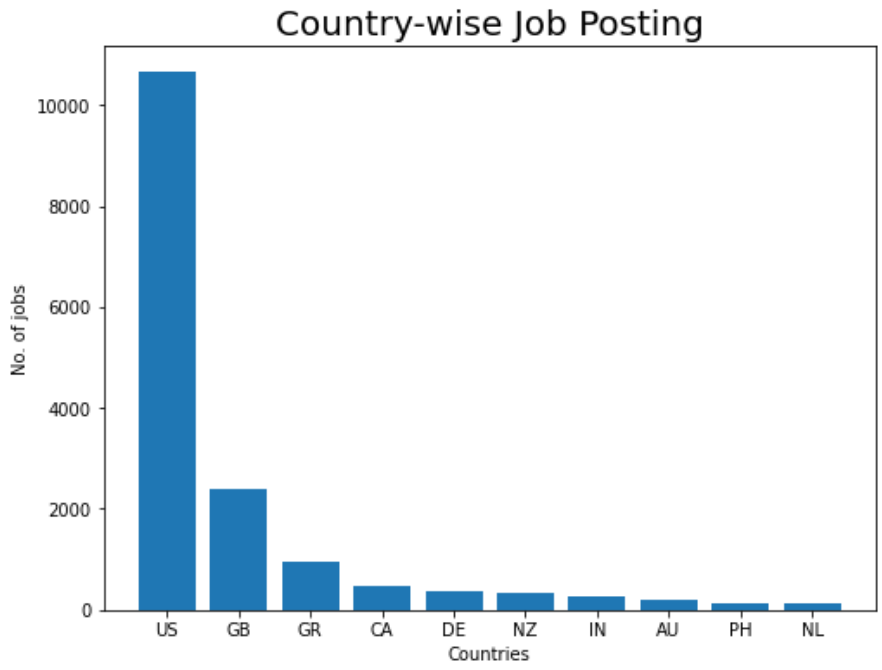


Figure 8.1 country wise job posting

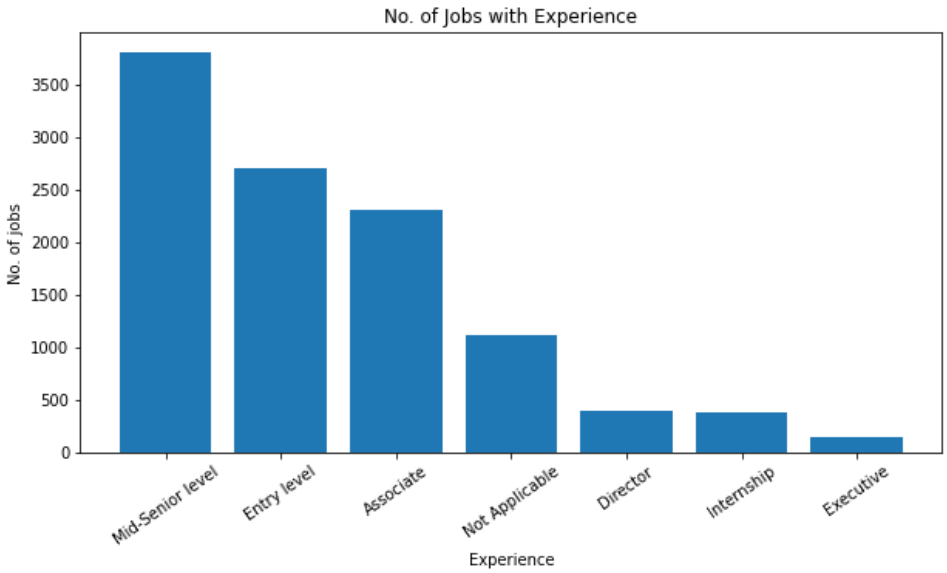


Figure 8.2 jobs with experience

8.2 Performance Evaluation

The mentioned classifier is trained and tested for detecting fake job posts over a given dataset that contains both fake and legitimate posts. It gives us 97% Average Accuracy.

```
In [22]:  1 print("Classification Accuracy:", accuracy_score(y_test, y_pred))

Classification Accuracy: 0.9656972408650261
```

8.3 Classification Report

This contains all results of model having precision, recall and f1-score.

```
In [22]:  1 print("Classification Report\n")

Classification Accuracy: 0.9656972408650261
Classification Report

              precision    recall  f1-score   support

           0       0.97      1.00      0.98      5094
           1       0.99      0.32      0.49       270

    accuracy                           0.97      5364
   macro avg       0.98      0.66      0.73      5364
weighted avg       0.97      0.97      0.96      5364
```

8.4 Confusion matrix

A confusion matrix is a matrix representation of the prediction results of any binary testing that is often used to describe the performance of the classification model (or "classifier") on a set of test data for which the true values are known.
The confusion matrix itself is relatively simple to understand, but the related terminology can be confusing.

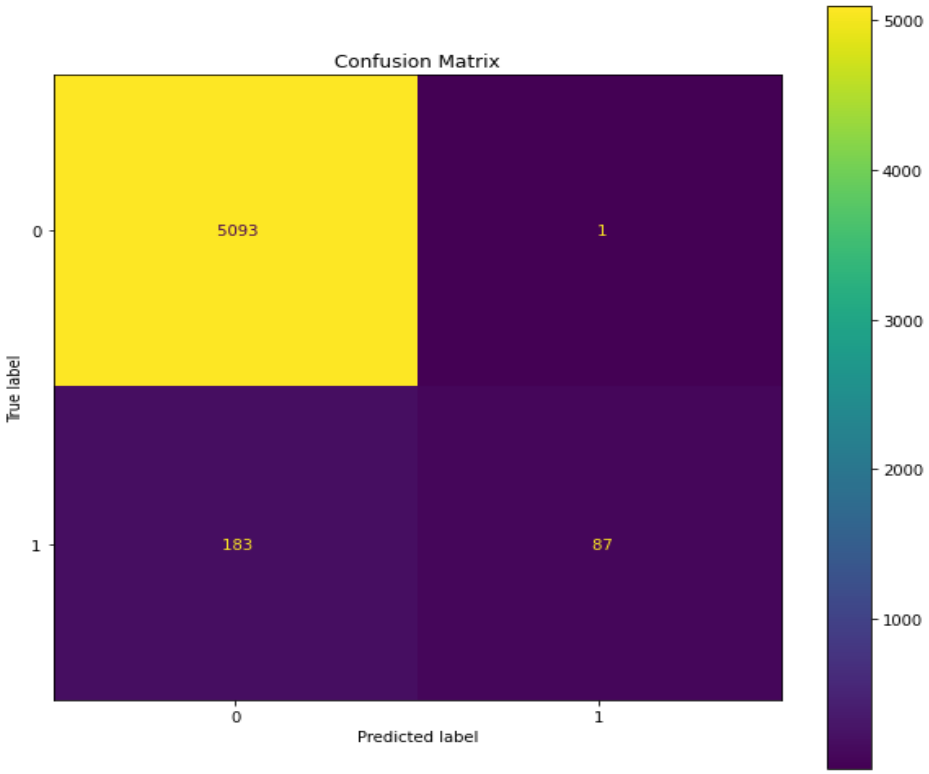


Figure 8.3 confusion matrix

8.5 Chapter Summary

In this chapter, I conclude all work of project tested and results after develop. In different ways it is tested to check that there is no error and calculate model accuracy and its results.

References

[1] B. A. a. F. Alharby, “An Intelligent Model for Online Recruitment Fraud Detection,” 2019.

[2] I. C. G. F. G. G. a. F. R. B. Biggio, “Bagging classifiers for fighting poisoning attacks in adversarial classification tasks,” 2011.

[3] P. C. a. S. J. Delany, “K -Nearest Neighbour Classifiers,” 2007.

[4] J. S. B. H. C. S. M. A. A. O. A. a. O. E. A. E. G. Dada, “Machine learning for email spam filtering: review, approaches and open research problems,,” 2019.

[5] A. N. a. A. Knoll, “Gradient boosting machines a tutorial,” p. vol. 7, DEC, 2013.

[6] H. S. a. S. Kumar, “A Survey on Decision Tree Algorithms of Classification in Data Mining,” 2016.

[7] F. Murtagh, “Multilayer perceptrons for classification and regression,,” 1991.

[8] H. T. M. G. R. I. H. a. M. K. N. Hussain, “Spam review detection techniques: A systematic literature review,” 2019.

[9] I. Rish, “ —An Empirical Study of the Naïve Bayes Classifier An empirical study of the naive Bayes classifier,,” January 2001.

[10] D. E. Walters, “Bayes’s Theorem and the Analysis of Binomial Random Variables,” 1988.