RESUME SCREENING USING MACHINE LEARNING

Project report submitted in partial fulfillment of the requirement for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by

V.KIRAN	P.ROHIT SAI	Y.AARTHI
(196C1A05A6)	(196C1A0578)	(196C1A0573)

Under the Esteemed Guidance of

Assoc.Prof - Mr.Atchyuta Rao

Department of Computer Science and Engineering

Miracle Educational Society Group Of Institutions

Accredited by NAAC, Permanently Affiliated to JNTUK

Approved by A.I.C.T.E

Bhogapuram, Vizianagaram District Dist.535216

2019-2023

MIRACLE EDUCATIONAL SOCIETY GROUP OF INSTITUTIONS

Miracle city, Bhogapuram, Vizianagaram-535216

Certificate

This is to clarify that the **project work** entitled **“Resume Screening using Machine Learning”** has been jointly carried out by

NAME OF THE STUDENT	REGD.NO
V.KIRAN	(196C1A05A6)
P.ROHIT SAI	(196C1A0578)
Y.AARTHI	(196C1A0573)

Under my guidance is partial fulfillment of the requirements for the award of the degree of Bachelor of Technology on Computer Science and Engineering of Jawaharlal Nehru Technological University, Kakinada during the Academic year 2019-2023.

Assoc.Prof . Mr.Atchyuta Rao
Project Guide

Prof. Mr. G.RAJA SEKHARAM
Head of the Department

External Examination

ACKNOWLEDGEMENT

We sincerely thank the following distinguished personalities who have given their advice and support for the successful completion of the work.

We are deeply indebted to our most respected guide **Mr.Atchyuta Rao**, Associate Professor, Department of CSE, for her valuable and inspiring guidance, comments, suggestions, and encouragement.

We extend our sincere thanks to **Mr. G RAJASEKHARAM** Associate Prof. & Head of the Dept. of CSE for extending his cooperation and providing the required resources.

We would like to thank our beloved principal **Dr. A. ARJUN RAO** for providing the online resources and other facilities to carry out this work.

We would like to express our sincere thanks to the project coordination committee and our project coordination **Mr. Atchyuta Rao**, Associate Prof., Dept. of CSE for his helpful suggestions in presenting this document.

We extend our sincere thanks to all other teaching faculty and Non-Teaching staff of the department, who helped directly or indirectly for their cooperation and encouragement.

V.Kiran(196C1A05A6)

Y. Aarthi(196C1AO573)

P. Rohit Sai(196C1A0578)

DECLARATION

We declare that this project work is composed by ourselves, and that the work contained herein is our own except where explicitly stated otherwise in the text. This work was not submitted earlier at any other university or institute for the award of any degree.

WITH SINCERE REGARDS

V.Kiran (196C1A05A6)

Y. Aarthi (196C1A0573)

P. Rohit Sai (196C1A0578)

Abstract

In a typical service organization, professionals with a variety of technical skills and business domain expertise are hired and assigned to projects to resolve customer issues. This task of selecting the best talent among many others is known as Resume Screening. People spend hours writing and formatting the perfect resume hoping it to be read by a talent acquisition professional and, eventually, help them land a job interview. Unfortunately, around 75% of resumes submitted are never seen by a human eye. Due to the high number of applicants and resume submissions to job postings, manual resume screening processes become tedious, ineffective, and time-consuming for talent acquisition professionals. Therefore, standardized automated screening methods are necessary to categorize qualified from unqualified candidates based on their background, education, and professional experience faster, with more efficiency and more accurate results to streamline hiring processes.

This is a Machine learning project on Resume Screening using Python programming. The main aim of our project is to classify the resumes on the basis of known languages mentioned by the candidates.

Index

List of Contents

S.NO	**LIST OF FIGURES**	**PAGE NO**

LIST OF TABLES	PAGE NO

CHAPTER 1

1. INTRODUCTION

1.1 OVERVIEW OF THE PROJECT

Resume screening is still the most time-consuming part of recruiting: screening resumes is estimated to take up to 23 hours for just one hire. When a job opening receives 250 resumes on average and 75% to 88% of them are unqualified, it's no wonder the majority of talent acquisition leaders still find the hardest part of recruitment is screening the right candidates from a large applicant pool. Compounding the problem, a recent survey of talent acquisition leaders found that 56% will increase their hiring volume next year, but 66% of recruiting teams will either stay the same size or shrink.

In 2019, "doing more with less" will depend on a recruiter's ability to figure out how and where to effectively automate their workflow. Advances in recruitment technology have added automation to candidate sourcing with recruitment marketing and to candidate interviewing with video interviews. However, technological innovations to address the biggest pain point in recruiting—screening resumes—have been frustratingly absent until recently.

The recruitment of new employees and allocation of suitable work has always been a random approach for many multinational companies [MNC]. With the use of the data obtained from the submitted resumes, this research work aims to separate the resume levels by analyzing the expert data from the resumes. To accomplish this, this research work has used machine learning and the Naïve-Bayes methodology and it also attempts to find which process provides the best possible result for the segregation using a one vs. rest classifier. Furthermore, this research work

attempts to find the accuracy and performance of the proposed methodology and incorporate it in the IT firms and other regulations for the prevention of manual screening and establish a safe allocation of resources for the companies.

Resume screening and matching the appropriate person to the appropriate job with the appropriate technology has long been a challenge for many companies throughout the recruiting process, and it is a major cause in many individuals leaving their jobs due to lack of enthusiasm. But if there is a chance to allocate a person with some technical knowledge/interest will always help in nurturing a long-term employee for the organization and this will also decrease the number of career drop-outs in every sector.

Employee attrition and other factors which make the organization fall behind others are also decreasing with the implementation of the proposed model. The proposed approach will separate resumes in such a way that the people with relevant experience in particular technology and, as a result, it enables the recruiter to move forward with the recruiting process. This will also reduce HR's workload in terms of manual follow-up and resume classification by using their knowledge base. This will be a game-changer for start-ups and MNCs in terms of their hiring process for both freshers and professionals, resulting in an increased turnover and efforts

The process of hiring in most of the organizations is nominal but the allocation of technology/domain after hiring a resource is random, which is a most troublesome and problematic phase for most of the new employees as they will not be able to work in their field of interest. The random assignment is the traditional process used by all the organizations and later on some employees selected from college by conducting some special drives will be given the option to pursue a career in their dream technology/domain, which gives them the drive to learn more and move forward but the proposed model makes all the resources/people to work in their area of interest

1.2 FEASIBILITY STUDY

In this, we study various feasibilities where existing and software equipment were sufficient for completing the project. The economic Feasibility determines whether doing the project is economically beneficial. The outcome of the first phase was that the request and various studies are approved and it was decided that the project taken up will serve the end user. On developing a implementing this software saves a lot of amount and sharing of valuable time

- Economical Feasibility
- Technical Feasibility
- Social Feasibility

1.2.1 Economical Feasibility

The study is carried out to check the economic impact that the system will have on the organization. The number of funds poured into the research and development of the system is limited. The expenditure is justified.

Intangible benefits include improving the decision-making process, enhancing accuracy, becoming more competitive in customer service, maintaining a good business image, and increasing job satisfaction for employees by eliminating tedious tasks.

1.2.2 Technical Feasibility

This is carried out to check the technical feasibility that is, the technical requirements of the system. Any system developed must not have a high demand on available technical recourses. The developed System must have modest requirements and are required for implementing this system. Our project has modest technical requirements.

Technical feasibility evaluates the technical complexity of the expert system and often involves determining whether the expert system can be implemented with state-of-the-art techniques and tools. In the case of expert systems, an important aspect of technical feasibility is determining the shell in which the system will be developed. The shell used to develop an expert system can be an important determinant to its quality and makes it vital to the system's success. Although the desirable characteristics of an expert system shell will depend on the task and domain requirements, the shell must be flexible enough to build expert reasoning into the system effectively. It must also be easily integrated with existing computer-based systems. Furthermore, a shell providing a user-friendly interface encourages end users to use the system more frequently

1.2.3 Social Feasibility

The aspect of the study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not be threatened by the system. His/her level of confidence must be increased so that he/she is able to make some constructive criticism which is welcomed. An appraisal exercise intimately connected with the evaluation of environmental feasibility is the assessment of the project’s impact on the lives of people that live and work in the project’s area of influence.

The social impact analysis (or social feasibility assessment) can be a very important part of the general appraisal of PPP projects since many infrastructure initiatives cause severe adverse impacts on communities surrounding the site on which they are implemented.

Social impact analysis is an exercise aimed at identifying and analyzing such impacts in order to understand the scale and reach of the project's social impacts. It also ensures that these impacts are mitigated, to the extent possible, and fully considered in the green light decision.

Social impact analysis greatly reduces the overall risks of the project, as it helps to reduce resistance, strengthens general support, and allows for a more comprehensive understanding of the costs and benefits of the project.

However, social impact analysis can be expensive and time-consuming, so the full analysis process cannot be justified for all projects. At a minimum, all projects demand a review of project data at the Appraisal Phase, so as to identify if material social impacts exist. If they do, a full social impact analysis should be conducted

1.3 SCOPE

The proposed software product is Resume Screening using Machine Learning. One vs. rest and Naïve bayes algorithms are used. After applying these techniques one can get the output as total how many categories are there in the pool of resumes and it shows which category we have the most candidates know and then the company can plan that which position is suitable for which and they also have a clear idea what people are interest in these days and it help them with saving a lot of the time and work power also.

CHAPTER 2

LITERATURE SURVEY

There have been over 50000 online recruitment sites which ask the job applicant candidates to submit their resumes on their website. In some of these websites, classification techniques for screening the resumes are not even employed. It is the job of the company recruiter to go through all the candidate resumes manually. This task is unassumingly daunting for the recruiters to select the most capable candidates for the subsequent rounds of the hiring process. Meanwhile, some recruitment sites have implemented the intelligent concept of automatically rating or classifying the resumes given by the candidates for a particular job position. Some of these websites or web applications are Indeed, Monster.com, Adecco.com, Top resume, Ideal etc. The description of

some of these websites including their advantages and disadvantages has been given below in detail. If we discuss one of the case study websites, indeed, resumes can be uploaded by the job applicants on their profile. This opens up the avenue for the prospective job seekers to apply to various job openings in various companies. Initially, this happened to be a good approach since the recruiters did not feel the pressure to manually go through each and every resume.

This is mainly because the job applicants had a much skewed number as compared to today's burgeoning number of job applicants. Going through the resumes of all the candidates has become a living nightmare for the already overburdened recruiters. They have to spend a lot of their energy and precious time to go through each and every candidate resume for selecting only the most appropriate candidates for the subsequent rounds. On top of this, the already frustrated recruiters learn that about 75% of the resumes submitted for the job opening have totally irrelevant skills for the given job description. Nevertheless, this website is still very popular where people post their resumes with the hope of getting selected for the subsequent rounds in the hiring process.

Coming to another case study, a website called Top Resume has employed the usage of techniques like Natural Language Processing to analyze a prospective job seeker's resume. Here, the task of the candidate is to only upload their resume on the portal. With the help of Natural language Processing, only the text data is extracted from the resume and the strength of the candidate's profile is displayed in terms of percentage. Additional attributes, such as the percentage of the skills the candidate has according to the education, certifications, courses and work experience of the candidate feature. Are also disclosed to the candidate itself. No provision has been made for any job applicant to apply for a particular job opening on this website, nor does this website have the provision of providing the recruiter with a rank list of all the resumes according to the relevant skills for the particular job position. There are many other web applications available in the literature providing mostly similar

2.1 EXISTING SYSTEM AND DRAWBACKS

The process of hiring in most organizations is nominal but the allocation of technology/domain after hiring a resource is random, which is a most troublesome and problematic phase for most of the new employees as they will not be able to work in their field of interest. The random assignment is the traditional process used by all organizations and later on some employees selected from college by conducting some special drives will be given the option to pursue a career in their dream technology/domain, which gives them the drive to learn more and move forward but the proposed model makes all the resources/people to work in their area of interest. To this day any company is not using a fully automated way to choose which is perfect for which position.

2.2 PROPOSED SYSTEM AND ADVANTAGES

The main aim of our project is to classify resumes based on their category. Our work is different from that of earlier proposed systems, as in most of the existing systems a job is recommended to the candidates based on their resume content, which leads to low classification accuracy. In order to improve it, we proposed a system that works by classifying the resume in their classes.

The first step is to clean our resumes and removed all the unnecessary data in the resumes and the next step is to classify the entire dataset based on the categories available in the dataset.

In the result, we display the visualization of the categories and which category is high in the resume. It will help the companies in both time and manpower management and it is an effective process for hiring people

Training set	Testing set

Table 1: Re-constructing dataset

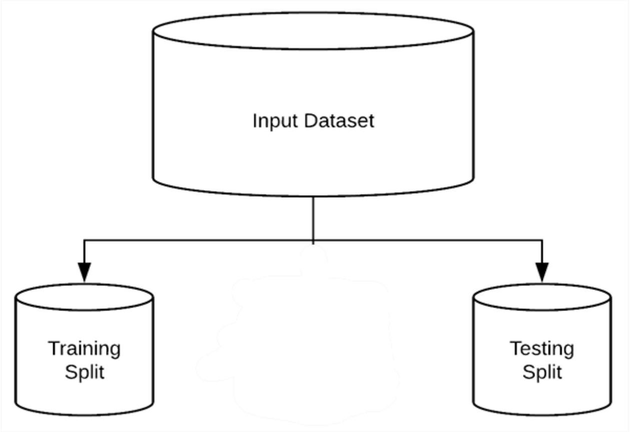


Fig 1: Split Dataset

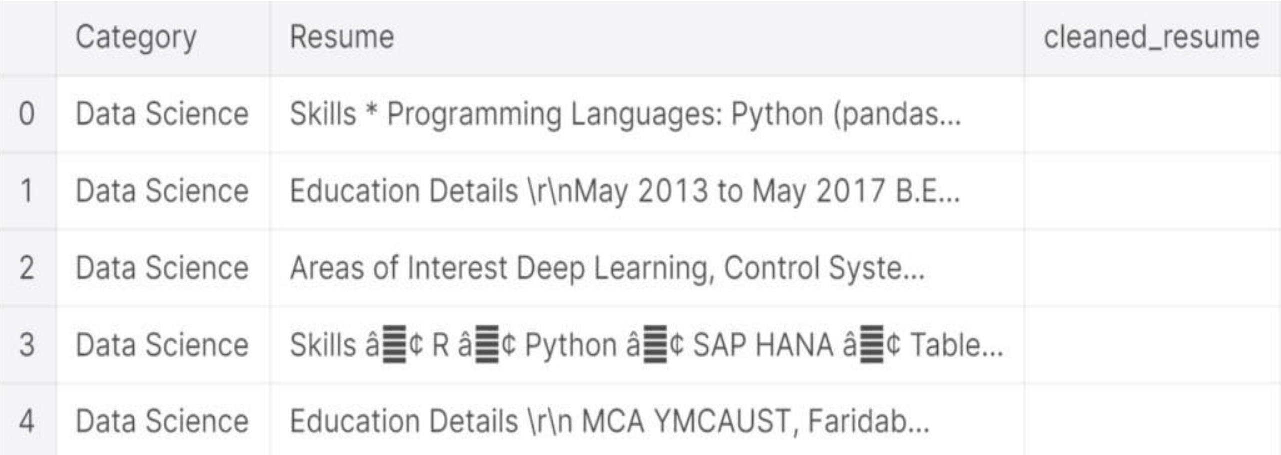
Dataset:

Data set Description

→ The data was downloaded from the online portal(s) and from Kaggle.

→ The data is in Excel format, with three column ID, Category, and Resume. ID

→ The sequence number of the resume, Category - Industry sector to which the resume belongs to, and Resume - The complete CV of the candidate.



	Category	Resume	cleaned_resume
0	Data Science	Skills * Programming Languages: Python (pandas...	
1	Data Science	Education Details \r\nMay 2013 to May 2017 B.E...	
2	Data Science	Areas of Interest Deep Learning, Control Syste...	
3	Data Science	Skills â≣¢ R â≣¢ Python â≣¢ SAP HANA â≣¢ Table...	
4	Data Science	Education Details \r\n MCA YMCAUST, Faridab...	

Fig 2: Dataset

CHAPTER 3

SYSTEM ANALYSIS

3.1 OVERVIEW OF SYSTEM ANALYSIS

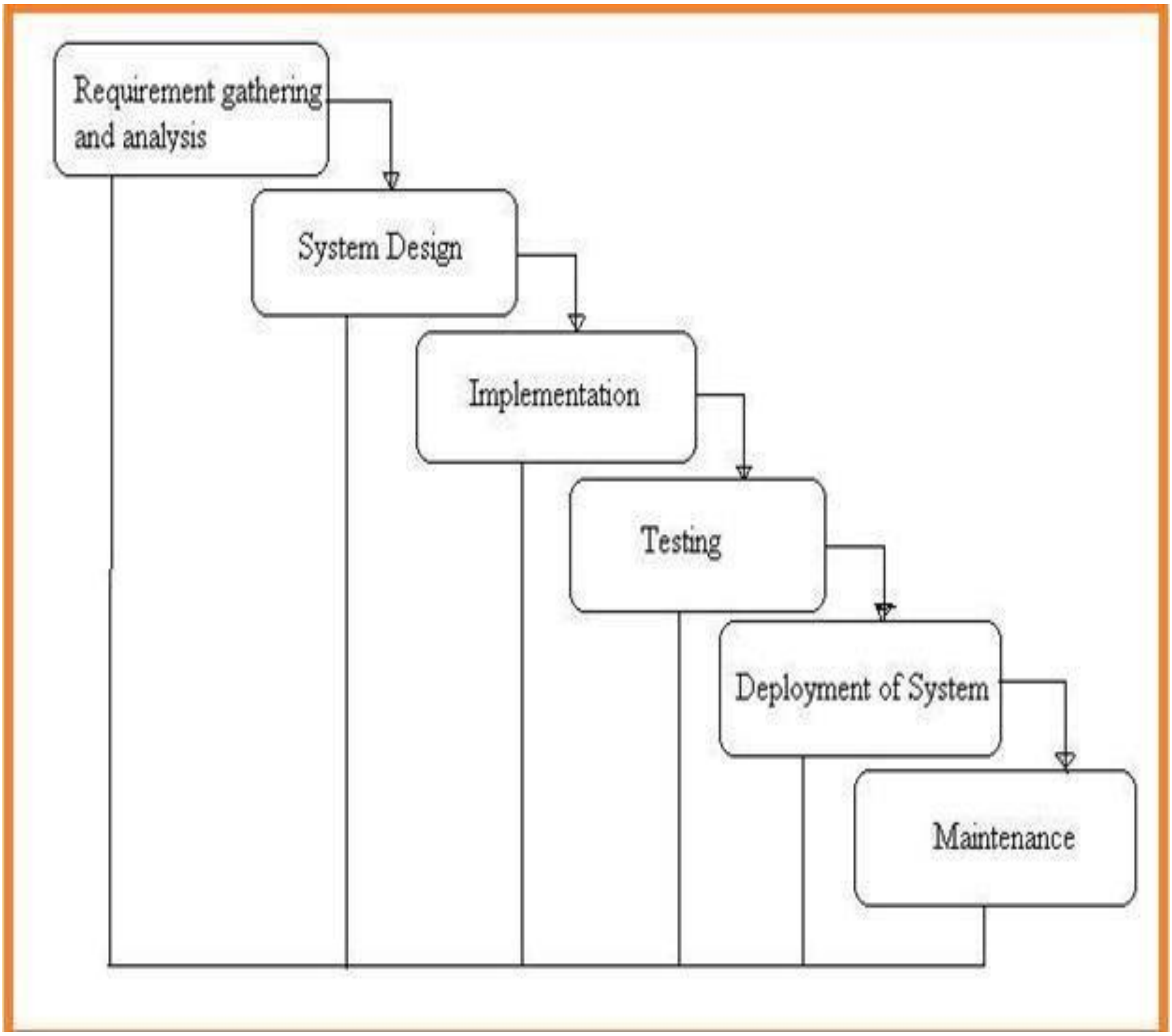


Fig 3: Project SDLC

- Project Requisites Accumulating and Analysis
- Application System Design
- Practical Implementation
- Manual Testing of My Application
- Application Deployment of System
- Maintenance of the Project

3.1.1 Requisites Accumulating and Analysis

It’s the first and foremost stage of any project as our is an academic leave for requisites amassing we followed IEEE Journals and Amassed so many IEEE Related papers and final culled a Paper designated by setting and substance importance input and for the analysis stage we took referees from the paper and did a literature survey of some papers and amassed all the Requisites of the project in this stage.

3.1.2 System Design

System Design has divided into three types GUI Designing, UML Designing with avails in development of the project in a facile way with a different actor and its utilizer case by utilizer case diagram, the flow of the project utilizing sequence, and the Class diagram gives information about different class in the project with methods that have to be utilized in the project if comes to our project our UML Will utilizable in this way The third and post-import for the project in system design is Data base design where we endeavor to design a database predicated on the number of modules in our project

3.1.3 Implementation

The Implementation is the Phase where we endeavor to give the practical output of the work done in the designing stage and most of the Coding in Business logic lay comes into action in this stage its main and crucial part of the project.

3.1.4 Testing

Unit Testing

It is done by the developer itself in every stage of the project and fine-tuning the bug and module is predicated additionally done by the developer only here we are going to solve all the runtime errors

Manual Testing

As our Project is on academic Leave we can do any automatic testing so we follow manual testing by endeavor and error methods

3.1.5 Deployment of System

Once the project is total we will come to the deployment of the client system in the genuine world as its academic leave we did deployment in our college lab only with all needed Software with having Ubuntu OS.

3.1.6 Maintenance

The Maintenance of our Project is a one-time process only.

3.2 Software Used in Project

What things you need to install the software and how to install them:

1. Python 3.8.8

   - This setup requires that your machine has python 3.8.8 installed on it. you can refer to this url https://www.python.org/downloads/ to download python. Once you have python downloaded and installed, you will need to setup PATH variables (if you want to run python program directly, detail instructions are below in *how to run software section*). To do that check this: https://www.pythoncentral.io/add-python- to-path-python-is-not-recognized-as-an-internal-or-external-command/.

   o Setting up PATH variable is optional as you can also run program without it and more instruction are given below on this topic.

2. Second and easier option is to download jupyter notebook on command prompt using the command pip install jupyter notebook (Optional).

3. You will also need to download and install below packages after you install python and jupyter notebook from the steps above

o numpy

o matplotlib

o sklearn (scikit-learn)

o Pandas

• Use below commands in command prompt to install these packages

```
pip install numpy
pip install matplotlib
pip install –U scikit-learn
pip install pandas
```

3.3 MODULES

Importing of the packages:

In this we will import all the packages that we are required for the project and ignore all the warnings and create a folder for the cleaned resumes in the dataset.

KNN

It is supervised technique, used for classification. "K" in the KNN repersent the number of nearest neighbours used to classify or predict in case of continuous variable.

NLP

NLP is a field in machine learning with the ability of a computer to understand, analyze, manipulate, and potentially generate human language.

The modules that we import in this is:

1. numpy
2. pandas
3. matplotlib
4. warnings (to remove all the warnings)

- NumPy

- NumPy is one of the fundamental packages for Python providing support for large

multidimensional arrays and matrices

- Pandas

- It is an open-source, Python library. Pandas enable the provision of easy data

structure

and quicker data analysis for Python. For operations like data analysis and

modelling,

- Matplotlib

- This open-source library in Python is widely used for publication of quality figures

in a variety of hard copy formats and interactive environments across platforms. You

can design charts, graphs, pie charts, scatterplots, histograms, error charts, etc. with

just a few lines of code.

Seaborn

When it comes to visualization of statistical models like heat maps, Seaborn is

among the reliable sources. This Python library is derived from Matplotlib and

closely integrated with Pandas data structures.

- Scipy

- This is yet another open-source software used for scientific computing in Python.

Apart from that, Scipy is also used for Data Computation, productivity, and high-performance computing and quality assurance

- Scikit-learn

It is a free software machine learning library for the Python programming language and can be effectively used for a variety of applications which include classification, regression, clustering, model selection, naive Bayes', grade boosting, K-means, and preprocessing.

- Nltk

Natural Language toolkit or NLTK is said to be one among the popular Python NLP Libraries. It contains a set of processing libraries that provide processing solutions for numerical and symbolic language processing in English only.

Unique categories:

Here we display all the unique categories in all the resumes. This help us get the idea of what evercategories are present in all the resumes. And then we count that how many resumes are there with the each category and produce an integer number for each of the category.

Visualization of Categories:

For the visualization in our project we choose the seaborn to visualize the result that we get inthe project then we create a bar graph based on the count of the all values of the all categories presented in the dataset.

Cleaning of the dataset:

Cleaning of the dataset means that we have to clean all the impurity from the dataset we choosethe regular expressions to clean the dataset.

```
import re
```

Then we send all this cleaned dataset to the cleaned dataset category in the dataset we created earlier.

Training and testing:

After sending all the words into the world cloud we use vectorization to assign a unique numbers And at the end we choose 0.8 of the data for the training and 0.2 data to the testing

We can always add the additional data.

3.4 SYSTEM REQUIREMENTS

3.4.1 SOFTWARE REQUIREMENTS

- Programming language : Python’s
- Version : 3.8.8
- Numpy, matplotlib, sklearn (scikit-learn),Pandas
- operating system: windows

Python:

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

AI and machine learning:

Because Python is such a stable, flexible, and simple programming language, perfect forvarious machine learning (ML) and artificial intelligence (AI) projects. In fact, Python is among the favourite languages among data scientists, and there are many Python machine learning and AI libraries and packages available.

Data analytics:

Much like AI and machine learning, data analytics is another rapidly developing field that utilises Python programming. At a time when we’re creating more data than ever before, there is a need for those who can collect, manipulate and organise the information.

Python for data science and analytics makes sense. The language is easy-to-learn, flexible, and well-supported, meaning it’s relatively quick and easy to use for analysing data.

Data Visualization:

Data visualisation is another popular and developing area of interest. Again, it plays into many of the strengths of Python. As well as its flexibility and the fact it’s opensource, Python provides a variety of graphing libraries with all kinds of features.

Whether you’re looking to create a simple graphical representation or a more interactive plot, you can find a library to match your needs. Examples include Pandas Visualization and Plotly. The possibilities are vast, allowing you to transform data into meaningful insights.

If data visualisation with Python sounds appealing, check out our 12-week ExpertTrack on the subject. You’ll learn how to leverage Python libraries to interpret and analyse data sets.

Programming applications:

You can program all kinds of applications using Python. The general-purpose language can be used to read and create file directories, create GUIs and APIs, and more. Whether it is blockchain applications, audio and video apps, or machine learning applications, you can build them all with Python.

We also have an Expert Track on programming applications with Python, which can help to kick-start your programming career. Over the course of 12 weeks, you’ll gain an introduction on how to use Python, and start programming your own applications using it.

Web development :

Python is a great choice for web development. This is largely due to the fact that there are many Python web development frameworks to choose from, such as Django, Pyramid, and Flask. These frameworks have been used to create sites and services such as Spotify, Reddit and Mozilla.

Thanks to the extensive libraries and modules that come with Python frameworks, functions such as database access, content management, and data authorisation are all possible and easily accessible. Given its versatility, it’s hardly surprising that Python is so widely used in web development

Game development :

Although far from an industry-standard in game development, Python does have its uses in the industry. It’s possible to create simple games using the programming language, which means it can be a useful tool for quickly developing a prototype. Similarly, certain functions (such as dialogue tree creation) are possible in Python.

If you’re new to either Python or game development, then you can also discover how to make a text-based game in Python. In doing so, you can work on a variety of skills and improve your knowledge in various areas.

Language development:

The simple and elegant design of Python and its syntax means that it has inspired the creation of new programming languages. Languages such as Cobra, Coffee Script, and Go all use a similar syntax to Python.

This fact also means that Python is a useful gateway language. So, if you’re totally new to programming, understanding Python can help you branch out into other areas more easily.

Finance:

Python is increasingly being utilised in the world of finance, often in areas such as quantitative and qualitative analysis. It can be a valuable tool in determining asset price trends and predictions, as well as in automating workflows across different data sources

NumPy:

NumPy is one of the fundamental packages for Python providing support for large multidimensional arrays and matrices

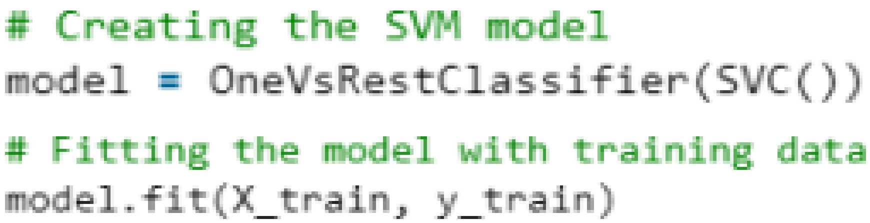
Pandas:

It is an open-source, Python library. Pandas enable the provision of easy data structure and quicker data analysis for Python. For operations like data analysis and modelling

Matplotlib:

This open-source library in Python is widely used for publication of quality figures in a variety of hard copy formats and interactive environments across platforms. You can design charts, graphs, pie charts, scatterplots, histograms, error charts, etc. with just a few lines of code

One-vs-Rest is a multi-class classification methodology used on the multiclass problems which can’t be solved by the traditional methodologies using the binary classification algorithms. The one-vs-rest methodology is offered by the python package scikit-learn (OneVsRestClassifier(estimator, *, n_jobs=None)). We can also use binary classifications inside the multi-class classification algorithms as follows:



```
# Creating the SVM model
model = OneVsRestClassifier(SVC())

# Fitting the model with training data
model.fit(X_train, y_train)
```

METHOLODIES

The naïve bayes algorithm used in the model is not a singular algorithm but a group of algorithms belonging to the similar purpose and process. In the process every feature selected from the data has an equal and independent contribution to the process for the outcome. Below mentioned is the bayes theorem equation which is applied to the data as follows

$$PX= P(X|y)P(y)P(X)$$

In the above equation we scope the variable x as the features and variable y as the class variable where the variable x is of multiple instances as followsX=(x1,x2,x3,….,xn). Ifthere is a case in which all the events are not dependent on each other, then the equation is

$$P(A,B) = P(A)\ P(B)$$

From the scenario we can deduce the equation as

$$\text{follows: } P(y|x1,\ldots,xn) =$$

$$Px1|yPx2|y\ldots.PyP(y)Px1Px2\ldots.P(xn)$$

The above expression can be represented as follows

$$P(y|x1,\ldots,xn) = P(y)i=1nP(xi|y)Px1Px2\ldots\ldots P(xn)$$

If we observe the expressions that are in denominator is a constant expression and it can be removed.

$$P(y|x1,\ldots, xn)\ \alpha\ Py\ i=1nP(xi|y)$$

We can now with all the above expressions derive a classifier process with all the requiredfeatures which give us the following expression.

$$Y=argmaxyP(y)\ i=1nP(xi|y)$$

3.4.2 HARDWARE REQUIREMENTS

- Processor: Intel core i3 and above
- RAM : Minimum 2GB
- HDD : Minimum of 10GB

CHAPTER 4

SYSTEM DESIGN

4.1 OVERVIEW OF SYSTEM DESIGN

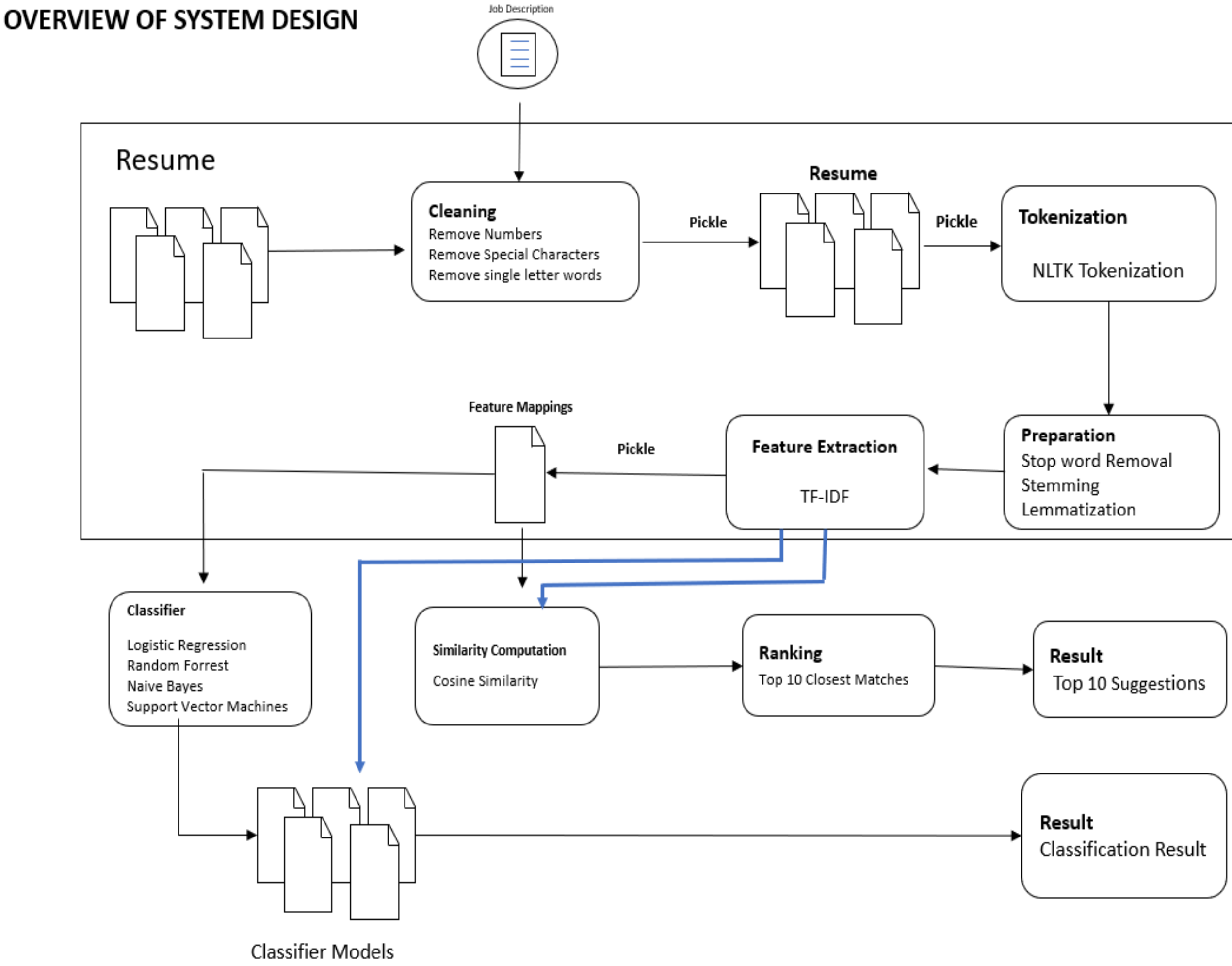


Fig: Architecture

4.2 UML DIAGRAMS

A UML diagram is a diagram based on the UML (Unified Modeling Language) with the purpose of visually representing a system along with its main actors, roles, actions, artifacts or classes, in order to better understand, alter, maintain, or document information about the system. As every diagram need not to be included in our project, we tested out what are the best suited

diagrams for our project. Some of them are,

1. Use case Diagram
2. Class Diagram
3. State Diagram
4. Sequence Diagram
5. Deployment Diagram

Use Case Diagram:

A usage case outline inside the Unified Modeling Language we used in our Project Development is Star (UML) could be a sort of behavioral chart portrayed out by and produced using a Use-case examination. Its inspiration is to gift a graphical layout of the presence of mind gave by a system to the extent performing specialists, their targets (addressed as use cases), and any conditions between those use cases. The most explanation behind a use case diagram is to show what structure limits are played out that on-screen character. Parts of the entertainers inside the system will be diagram.

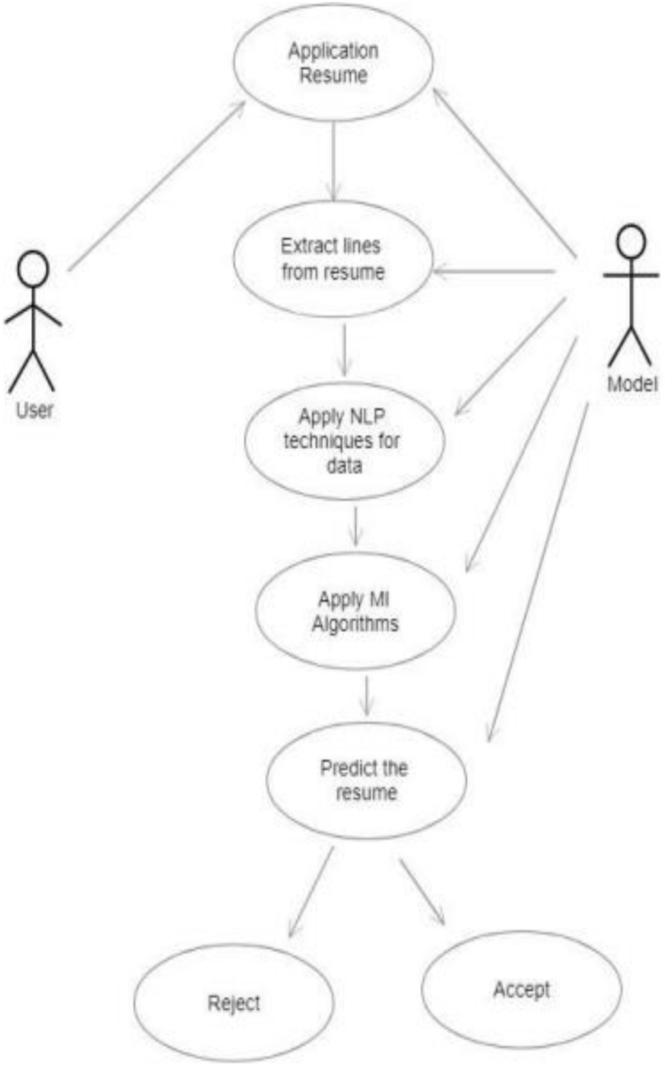


Fig 5: Use Case Diagram

Class Diagram:

In PC code planning, a class plot inside the Unified Modeling Language we used in or Project Development in star (UML) could be a sort of static structure outline the delineates the structure of a system by showing the systems' characterizations, their qualities, operations (or methodologies), and moreover the associations among the groupings. It elucidates that class contains data.

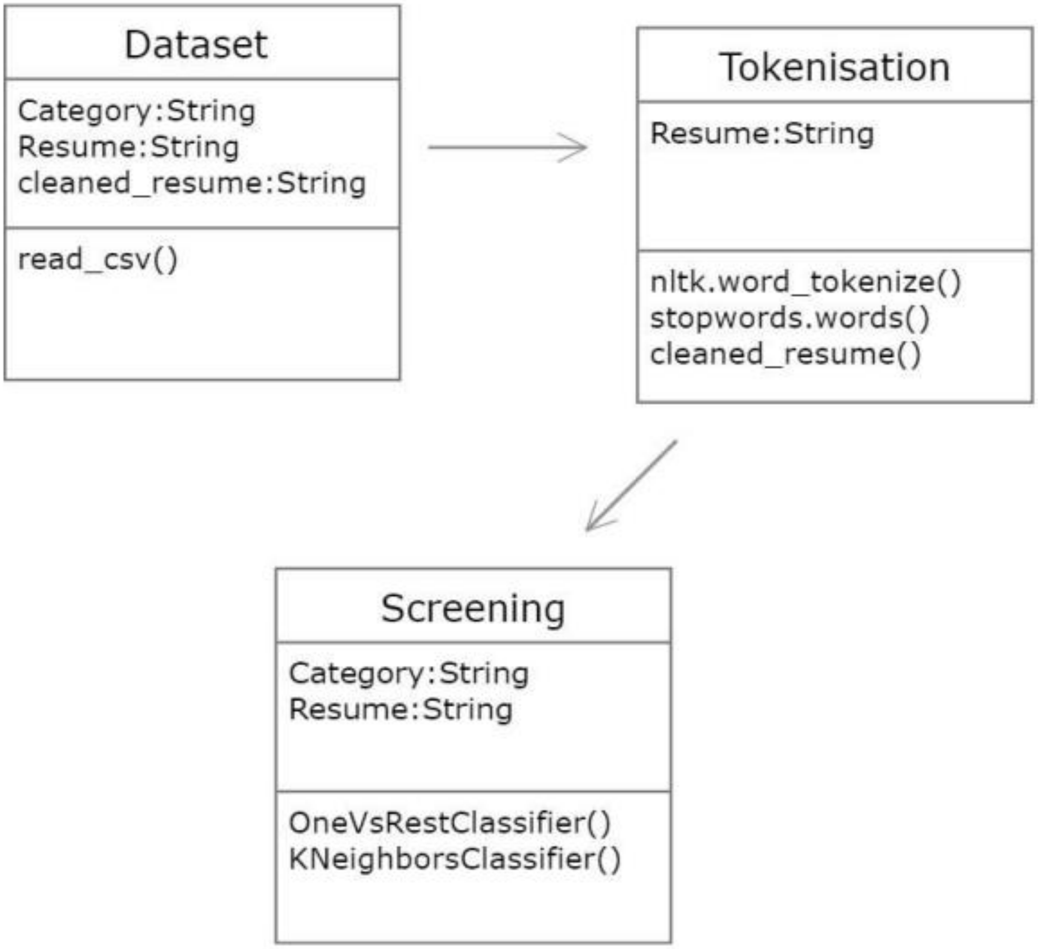


Fig 6: Class Diagram

State Diagram:

A state diagram is used to represent the condition of the system or part of the system at finite instances of time. It’s a behavioral diagram and it represents the behavior using finite state transitions

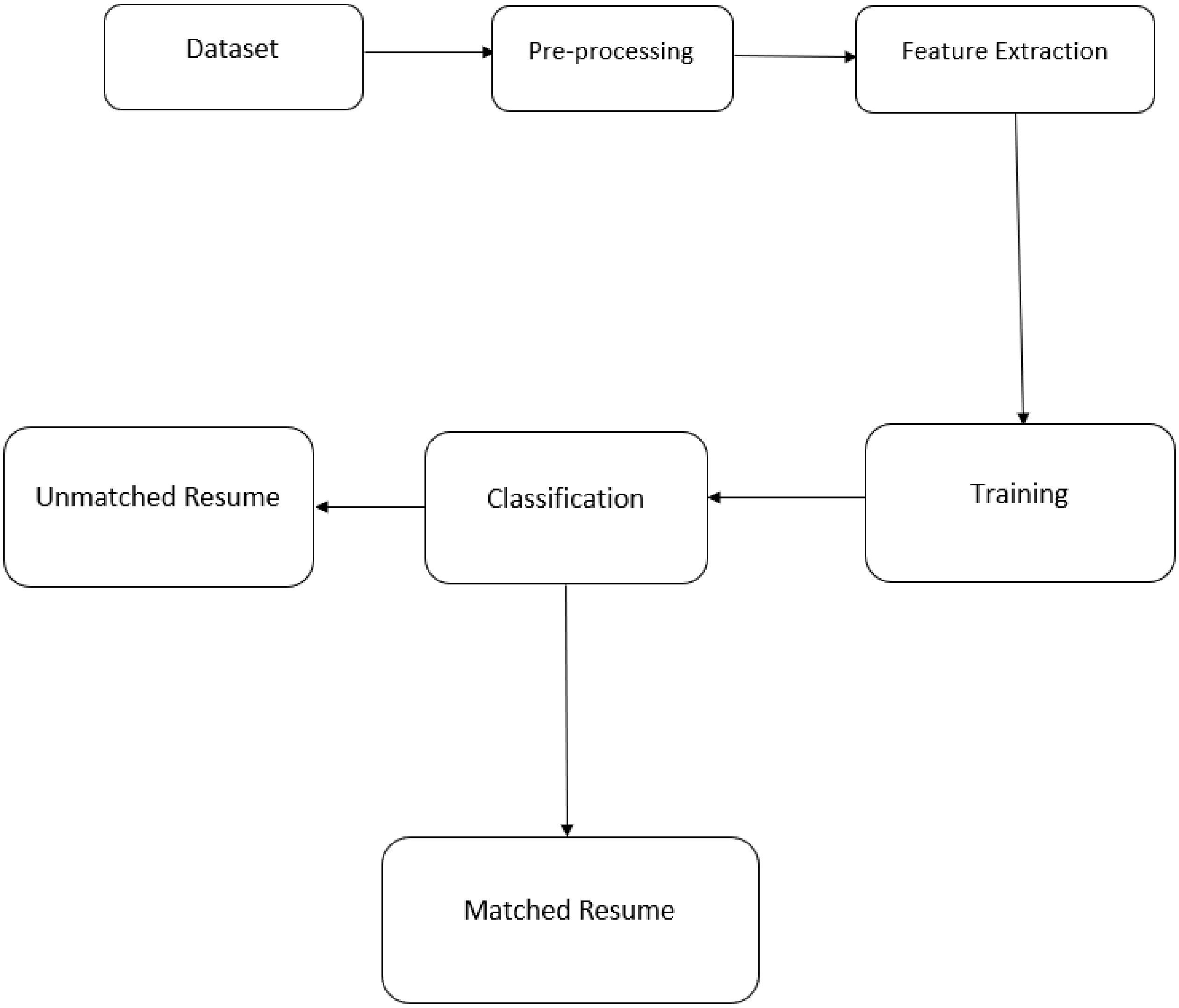


Fig 7: State Diagram

Sequence Diagram:

A sequence diagram shows the sequence of messages passed between objects. Sequence diagrams can also show the control structures between objects.

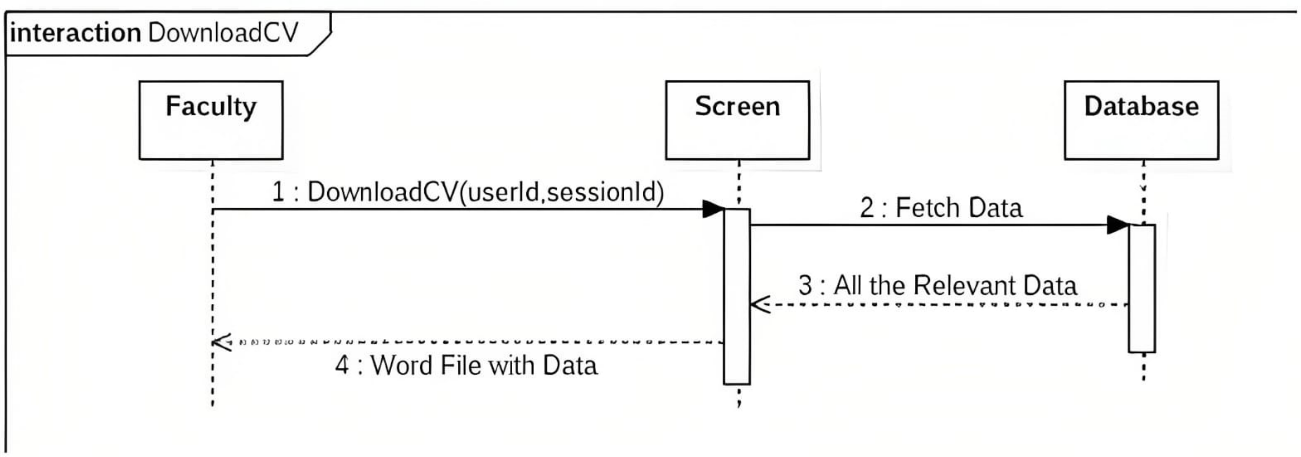


Fig 8: sequence Diagram

Deployment Diagram:

Deployment diagrams are used to visualize the topology of the physical components of a system, where the software components are deployed. Deployment diagrams are used to describe the static deployment view of a system. Deployment diagrams consist of nodes and their relationships.

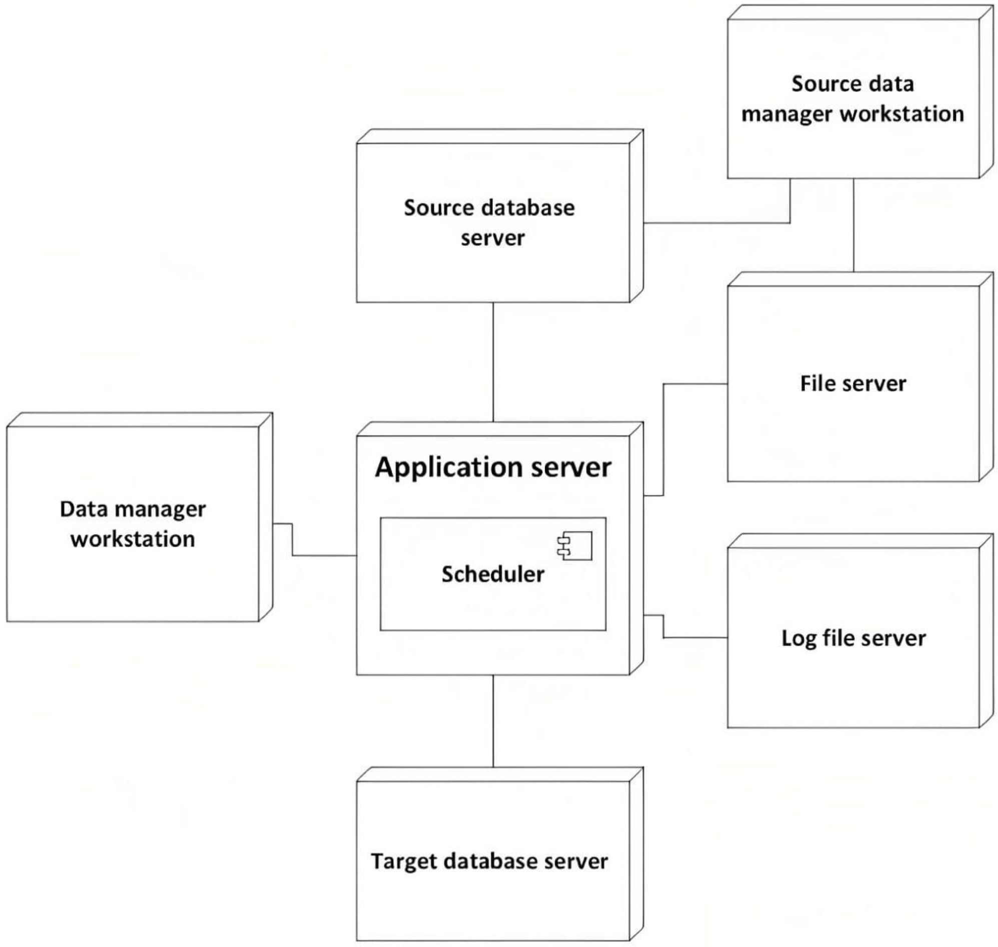


Fig 9: Deployment Diagram

CHAPTER 5

CODING & IMPLEMENTATION

resume_screeing.ipynb

```
import
numpy
as np
import
pandas
as pd
import
matplotlib.pyplot as
plt import warnings
warnings.filterwarnin
gs('ignore')
from sklearn.naive_bayes import
MultinomialNB from sklearn.multiclass
import OneVsRestClassifier from sklearn
import metrics
from sklearn.metrics import
accuracy_scorefrom
pandas.plotting import
scatter_matrix
```

```
from sklearn.neighbors import
KNeighborsClassifierfrom sklearn
import metrics
resumeDataSet=pd.read_csv('UpdatedResumeDataSet.csv',encoding='ut8')
resumeDataSet['cleaned_resume'] = "
resumeDataSet.head()
print ("Displaying the distinct categories of resume -")print
(resumeDataSet['Category'].unique())
```

```
print ("Displaying the distinct categories of resume and the number of records belonging to each category -")

print (resumeDataSet['Category'].value_counts())

pip install seaborn

import seaborn as sns

plt.figure(figsize=(15,15))

plt.xticks(rotation=90)

sns.countplot(y="Category", data=resumeDataSet)

from matplotlib.gridspec import GridSpec

targetCounts = resumeDataSet['Category'].value_counts()

targetLabels = resumeDataSet['Category'].unique()

# Make square figures and axes

plt.figure(1, figsize=(25,25))

the_grid = GridSpec(2, 2)

cmap = plt.get_cmap('coolwarm')

colors = [cmap(i) for i in np.linspace(0, 1, 3)]

plt.subplot(the_grid[0,1],aspect=1,title='CATEGORYDISTRIBUTION')
```

```
 source_pie=plt.pie(targetCounts,labels=targetLabels,autopct='%1.1f%%',shadow=True,
colors=colors
plt.show()
import re
def cleanResume(resumeText):
resumeText = re.sub('http\S+\s*', ' ', resumeText) # remove URLs
resumeText = re.sub('RT|cc', ' ', resumeText) # remove RT and cc
resumeText = re.sub('#\S+', '', resumeText) # remove hashtags
resumeText = re.sub('@\S+', ' ', resumeText) # remove mentions
resumeText = re.sub('[%s]' % re.escape("""!"#$%&'()*+,-./:;<=>?@[\]^_`{|}~"""), ' ',
resumeText) # remove punctuations
resumeText = re.sub(r'[^\x00-\x7f]',r' ', resumeText)
resumeText = re.sub('\s+', ' ', resumeText) # remove extra whitespace
return resumeText
resumeDataSet['cleaned_resume']=resumeDataSet.Resume.apply(lambdax:clean
Resume(x))
import nltk
nltk.download("punkt")
import nltk
from nltk.corpus import stopwords
import string
```

```
from wordcloud import WordCloud
oneSetOfStopWords = set(stopwords.words('english')+['``',"''"])
totalWords =[]
Sentences = resumeDataSet['Resume'].values
cleanedSentences = ""
for i in range(0,160):
cleanedText = cleanResume(Sentences[i])
cleanedSentences += cleanedText
requiredWords = nltk.word_tokenize(cleanedText)
for word in requiredWords:
if word not in oneSetOfStopWords and word not in string.punctuation:
totalWords.append(word)
wordfreqdist = nltk.FreqDist(totalWords)
mostcommon = wordfreqdist.most_common(50)
print(mostcommon)
wc = WordCloud().generate(cleanedSentences)
plt.figure(figsize=(15,15))
plt.imshow(wc, interpolation='bilinear')
plt.axis("off")
plt.show()
from sklearn.preprocessing import LabelEncoder
var_mod = ['Category']
le = LabelEncoder()
for i in var_mod:
resumeDataSet[i] = le.fit_transform(resumeDataSet[i])
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from scipy.sparse import hstack
requiredText = resumeDataSet['cleaned_resume'].values
requiredTarget = resumeDataSet['Category'].values
word_vectorizer = TfidfVectorizer(
sublinear_tf=True,
stop_words='english',
max_features=1500)
word_vectorizer.fit(requiredText)
WordFeatures = word_vectorizer.transform(requiredText)
```

```
print ("Feature completed .....")
X_train,X_test,y_train,y_test=train_test_split
(WordFeatures,requiredTarget,random_state=0,
test_size=0.2)
print(X_train.shape)
print(X_test.shape)
print(WordFeatures)
clf = OneVsRestClassifier(KNeighborsClassifier())
clf.fit(X_train, y_train)
prediction = clf.predict(X_test)
print('Accuracy of KNeighbors Classifier on training set: {:.2f}'.format(clf.score(X_train,
y_train)))
print('Accuracy of KNeighbors Classifier on test set: {:.2f}'.format(clf.score(X_test,
y_test)))
print("\n Classification report for classifier %s:\n%s\n" % (clf,
metrics.classification_report(y_test, prediction)))
```

IMPLEMENTATION

As we can see that our best performing models had an accuracy in the range of 99 and f1 score in the range of 99’s. This is due to less number of data that we have used for training purposes and simplicity of our models. In addition, we could also increase the training data size.We will extend this project to implement these techniques in future to increase the accuracy and performance of our models.

Resume screening is a strategy largely used by Big Tech businesses to sort through a huge number of resumes, rank them according to resume strength or relevance to the job description, and then filter them. The student seeking for the position, on the other hand, has no understanding why his resume was turned down or how he might modify his CV to make it more relevant and remarkable.

There is no technology available right now that would benefit students and help them create their resume. To resolve the given issue statement, the machine learning model will be employed. It will read the student's resume and extract information such as abilities and credentials. It also takes connections to the student's GitHub and LinkedIn profiles for more information, from which it can extract the student's contributions in a variety of fields. The student must also state which job role he or she is applying for. The model is trained using a set of job descriptions and skill sets.

CHAPTER 6

SYSTEM TESTING

6.1 OVERVIEW OF TESTING

Software Testing is evaluation of the software against requirements gathered from users and system specifications. Testing is conducted at the phase level in software development life cycle or at module level in program code. Software testing comprises of Validation and Verification.

6.2 TYPES OF TEST

6.2.1 UNIT TESTING

UNIT TESTING is a level of software testing where individual units/ components of a software are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest testable part of any software. It usually has one or a few inputs and usually a single output.

6.2.2 INTEGRATION TESTING

INTEGRATION TESTING is a level of software testing where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units. Test drivers and test stubs are used to assist in Integration Testing.

6.2.3 BLACK BOX TESTING

BLACK BOX TESTING is a software testing method in which the functionalities of software applications are tested without having knowledge of internal code structure, implementation details and internal paths. Black Box Testing mainly focuses on input and output of software applications and it is entirely based on software requirements and specifications. It is also known as Behavioral Testing.

6.2.4 WHITE BOX TESTING

WHITE BOX TESTING is software testing technique in which internal structure, design and coding of software are tested to verify flow of input-output and to improve design, usability and security. In white box testing, code is visible to testers so it is also called Clear box testing, open box testing, transparent box testing, Code-based testing and Glass box testing.

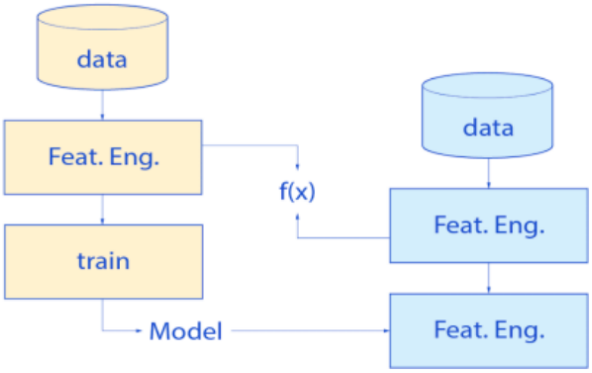
6.2.5 SYSTEM TESTING

SYSTEM TESTING is a level of testing that validates the complete and fully integrated software product. The purpose of a system test is to evaluate the end-to-end system specifications. Usually, the software is only one element of a larger computer-based system. Ultimately, the software is interfaced with other software/hardware systems. System Testing is actually a series of different tests whose sole purpose is to exercise the full computer-based system.

CHAPTER 7

RESULTS

On the resumes we now apply the models and get the results by implementing the Naïve-Bayes and one-vs-rest classifier algorithms and obtain the results by comparing them with each other. The accuracy and performance of the models are calculated. It is shown that, naïve-bayes has achieved the best results with 99% accuracy for all the models considered. The proposed approach deduce that, this can be further extended by incorporating the process to the organizational level hiring model and increase the individuals, who can work on their area of interest/expertise.



The pre-processing is done for the data and it is converted to a comma separated format file for further handling of data. Then the filling up of the inconsistent/missing data is done and normalization of data takes place which gives us the training set for further analysis. We used the Naïve-bayes and one-vs-rest classifier algorithms for processing the resumes and segregate accordingly. The graphical representation given above is the result obtained when the proposed model is implemented on a sample set of resumes and are segregated into batches based on their field of interest/expertise. The employee attrition and other factors which make the organization fall-behind others will also be decreasing with the implementation of the proposed model.

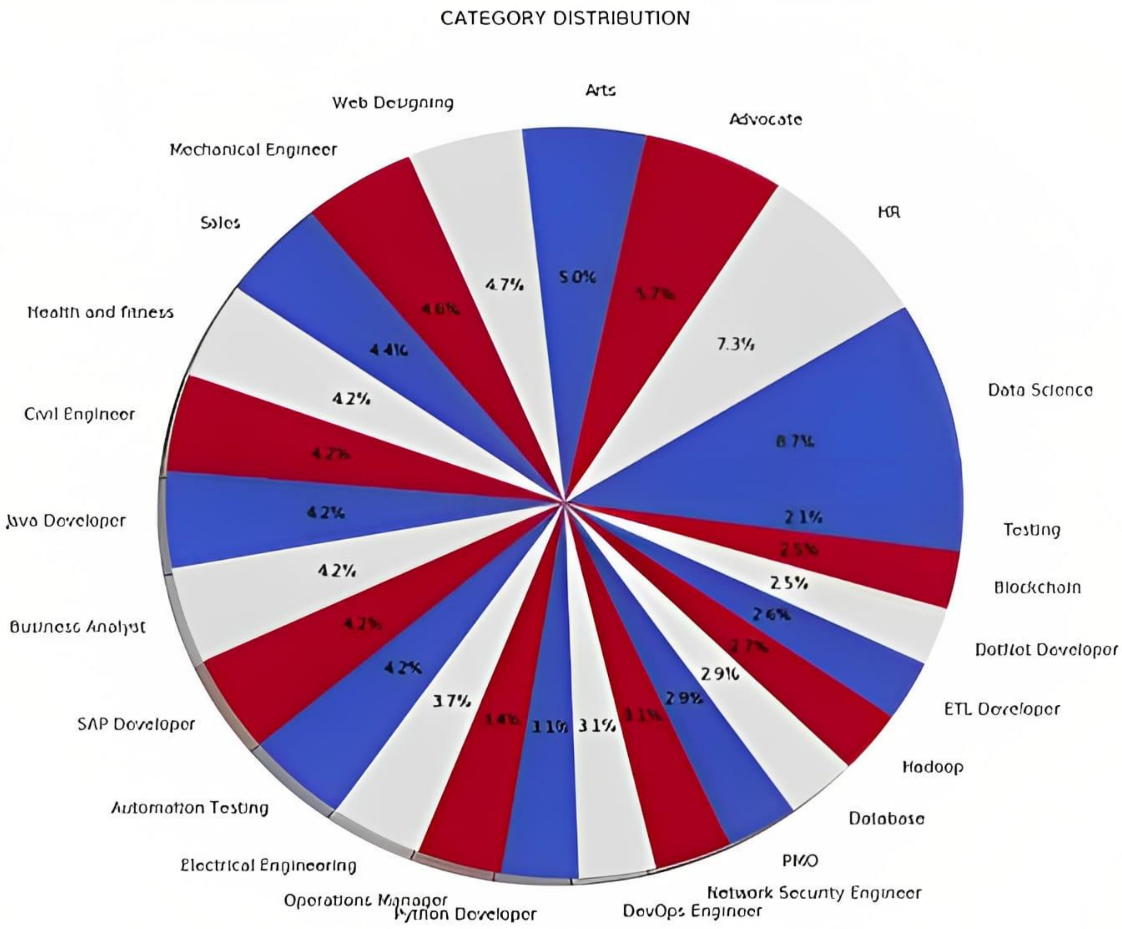


Fig 8 : Pie chart of resume classification

CHAPTER 8

CONCLUSION

Here with the implementation of Naïve-Bayies and onevsrest algorithms we achieve the resume screening and segregation. The resumes are segregated in a way such that we can see which person has knowledge on which technology and in implication to that we can move forward with the hiring-process. This will also lessen the work for the HR for manual follow-up and segregation of resumes by means of their knowledge base. Only A selected few from college itself by some special drives are given the option to pursue a career in their dream technology/domain which gives them the drive to learn more and move forward but the proposed model makes all the resources/people to work in their area of interest. We can further improve the method my incorporating more efficient and complex methodologies along with more tuning algorithms to enlarge the scope of benefit and lessen the risk factor for the employees

REFERENCES

[1] Pradeep Kumar Roy, Vellore Institute of Technology, 2019. A Machine learning approach for automation of resume recommendation system, ICCIDS 2019. 10.1016/j.procs.2020.03.284.

[2] Thimma Reddy Kalva, Utah State University, 2013. Skill-Finder: Automated Job-Resume.

[3] Based Framework for automatic resume quality Suhjit Amin, Fr.Conceicao Rodrigues Institute of Technology, 2019. Web Application for Screening resume, IEEE DOI: 10.1109/ICNTE44896.2019.8945869.

[4] Ashwini K, Umadevi V, Shashank M Kadiwal,Revanna, Design and Development of e Learning based Resume Ranking.

[5] Riza tana Fareed, rajah V, and Sharadadevi kaganumat "Resume Classification and Ranking using KNN and Cosine Similarity" In 2021 International Journal of Engineering.

[6] Sujit Amin, Nikita Jayakar, Sonia Sunny, Pheba Babu, M. Kiruthika, Ambarish Gurjar, Web Application for Screening Resume, 2019 International Conference on Nascent Technologies in Engineering (ICNTE), DOI: 10.1109/ICNTE44896.2019.8945869.

[7] Suhas H E, Manjunath AE, "Differential Hiring using Combination of NER and Word

Embedding”, In 2020 International Journal of Recent Technology and Engineering (IJRTE), ISSN: 2277-3878, Vol.9

[8] Centre for Monitoring Indian Economy Pvt Ltd. (CMIE),2022. The unemployment rate in India.

[9] Howard, J.L., Ferris, G.R., 1996. The employment interview context: Social and situational influences on interviewer decisions Xavier Schmitt, Sylvain Kubler, Jer my Robert, Mike Papadakis, Yves LeTraon University of Luxembourg, Luxembourg Replicable Comparison Study NER Software: StanfordNLP, NLTK, OpenNLP, SpaCy, Gate.

[10] Y. Luo, Y. Wen, T. Liu, and D. Tao, “Transferring knowledge fragments for learning distance metric from a heterogeneous domain,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018.

[11] Mikheev, Andrei; Moens, Marc; Glover, 1999. “Named Entity Recognition without Gazetteers.” Proceedings of EACL ’99. HCRCLanguage Technology Group, University of Edinburgh, http://acl.ldc.upenn.edu/E/E99/E99-1001.pdf.

[12] Al-Otaibi, S.T., Ykhlef, M., 2012. A survey of job recommender systems. International Journal of Physical Sciences 7, 5127–5142.

[13] Bhushan Kinge*1, Shrinivas Mandhare2, Pranali Chavan3, S. M. Chaware4 , Resume

Screening Using Machine Learning and NLP : AProposed System, International Journal of Scientific

Research in Computer Science, Engineering and

Information Technology, ISSN : 2456-3307 UGC

Journal No : 64718.

[14] Scholkopf, B., Smola, A.J., Bach, F., et al., 2002.

Learning with kernels: support vector machines,

regularization, optimization, and beyond, MIT press.