

# **Job Role Prediction**

A project report submitted in partial fulfillment  
of the requirements for the degree of

## **Bachelor of Engineering**

In

## **Computer Engineering**

Submitted by

Juhi Checker (Roll No: 7920)

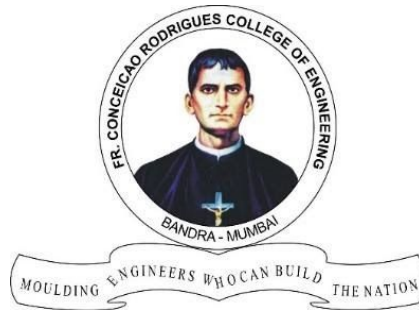
Sayali Deo(Roll No: 7926)

Anne Rajan (Roll No: 8160)

Under the guidance of

**Dr. Brijmohan Daga**

**(Head of Department, Computer Engineering Department)**



Department of Computer Engineering

Fr. Conceicao Rodrigues College of Engineering

Bandra, Mumbai-400 050

Year: 2019-2020

# **Internal Approval Sheet**

## **CERTIFICATE**

This is to certify that the project entitled "**Job Role Prediction**" is a bonafide work of **Juhi Checker(7920)**, **Sayali Deo (7926)**, **Anne Rajan(8160)** submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of "**Bachelor of Engineering**" in "**Computer Engineering**".

**Prof : Dr. Brijmohan Daga**

Supervisor/Guide

**Dr. Brijmohan Daga**

Head of Department

**Dr. Srijan Unnikrishnan**

Principal

## Project Report Approval

This project report entitled by ***Job Role Prediction*** by ***Juhi Checker, Sayali Deo, Anne Rajan*** is approved for the degree of **Bachelor of Engineering** in **Computer Engineering**.

Examiners

1.

—

2.

Date:

Place:

# Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Juhi Checker (Roll No. 7920)

Sayali Deo (Roll No. 7926)

Anne Rajan (Roll No. 8160)

Date: April , 2020

## **Acknowledgments**

We have great pleasure in presenting the report on "**Job Role Prediction**". We take this opportunity to express our sincere thanks towards the guide **Dr. Brijmohan Daga**, C.R.C.E, Bandra (W), Mumbai, for providing the technical guidelines, and the suggestions regarding the line of this work. We enjoyed discussing the work progress with his during our visits to the department.

We thank Dr. Brijmohan Daga, Principal and the management of C.R.C.E., Mumbai for encouraging and providing necessary infrastructure for pursuing the project.

We also thank all non-teaching staff for their valuable support, to complete our project.

Juhi Checker(Roll No. 7920)

Sayali Deo(Roll No. 7926)

Anne Rajan(Roll No. 8160)

Date:        April , 2020

# Contents

<b>Abstract</b>	<b>i</b>
<b>List of Diagrams</b>	<b>ii</b>
<b>1. Introduction</b>	<b>1</b>
1.1 Problem Statement	1
1.2 Motivation	2
1.3 Aim and Scope	2
<b>2. Literature Survey</b>	<b>3</b>
2.1 Previous Work in the field	3
2.2 Existing System	5
<b>3. Proposed System</b>	<b>6</b>
3.1 Proposed Solution	6
3.2 Methodology Overview	6
3.3 Architecture	7
<b>4. Implementation</b>	<b>9</b>
4.1 Dataset	9
4.1.1 Input Parameters	10
4.1.2 Output Parameters	11
4.2 Training Parameters	12
4.3 Code for Feature Extraction	13

4.4 Code for ANN	15
4.5 Code for Flask	16
<b>5. Technology Used</b>	<b>18</b>
5.1 Python	18
5.2 Visual Studio	18
5.3 Libraries used	18
5.3.1 Tensorflow	18
5.3.2 Flask	18
5.3.3 Scikit-learn	19
5.3.4 Keras	19
5.3.5 Jinja	19
5.3.6 SQLAlchemy	19
<b>6. Hardware and Software Requirements</b>	<b>20</b>
<b>7. Project Timeline</b>	<b>21</b>
<b>8. Result &amp; Conclusion</b>	<b>23</b>
8.1 Results Analysis	23
8.2 Conclusion	24
8.3 Snapshots	24
<b>9. References</b>	<b>31</b>

## **Abstract**

There is a trend amongst students to generally opt for career paths based on either the choices of their colleagues or the highest salary paying roles. They fail to know their strengths and choose their career randomly which leads to frustration and demoralization. Moreover, while recruiting the candidates, recruiters need to assess them in all different aspects. Thus, there is a need for a system that helps students decide a job role that is best suited for him/her which is based on his/her skill-set and other evaluation metrics which is now possible due to advancements in the field of deep learning. We propose an automated system using an Artificial Neural Network which considers personality traits of the individual along with personal interests and academics to predict which computer science job role would be best suited for them.



## List of Diagram

Figure 1 - ANN Flowchart.....	6
Figure 2 - ANN Architecture.....	7
Figure 3 - Cost Function.....	23
Figure 4 - Training and Testing Accuracy.....	23

# 1. Introduction

Artificial Neural Networks is a data processing system consisting of a large number of simple, highly interconnected processing elements in an architecture inspired by the structure of the cerebral cortex portion of the brain. Hence, neural networks are often capable of doing things which humans or animals do well but which conventional computers often do poorly. For career recommendation various parameters are considered which becomes quite difficult to predict using traditional regression models. In recent years, recommendation systems have been widely used in various commercial platforms to provide recommendations for users. Every field has various job roles which makes it challenging for any undergraduate student and recruiter to decide a well-suited job for students. Any student after graduation needs to decide which job role is best suited for him according to his profile. This is important for a long-term career plan. Similarly, for a recruiter it is very crucial to recruit a candidate after assessing him/her in all different aspects. A career recommender system will help undergraduate students and recruiters in finding the right job based on their personality, academics, interests, etc. We propose a job role prediction system using neural networks due to the high number of parameters for classification. These parameters include student performance in various subjects present in the undergraduate curriculum of computer science as well as student interests, interpersonal skills, talents, etc. The model is trained and tested on 15,000 and 3,000 dataset entries respectively. The model performs multiclass classification and is able to predict one of the 6 domains (i.e. Database Administrator, Project Manager, Software Developer, Business Intelligence Analyst, Security Administrator, Technical Support).

## 1.1 Problem Statement

The problem statement is ‘To analyze the Job-Role for Computer Engineering Students by taking certain parameters into consideration for e.g.(Academic Interests, Coding Skills, Hackathons, Personality etc.)’. The application will help an undergraduate to choose an ideal Job-Role suited for his profile. This is possible using Artificial Neural Networking. Moreover, it will help an undergraduate to build his resume and to prepare for those Job-Roles.

## **1.2 Motivation**

The awareness of choosing the right option for career is increasing among the students. The students fail to know their strengths and choose their career randomly which leads to frustration and demoralization. The automated system is used to evaluate the personality traits of the individual and predict which field would be best suited for them. The accuracy of prediction depends on the set of relevant skill parameters and analytics on these. The research in view to design the accurate prediction and recommendation systems is shown by Artificial Neural Network.

## **1.3 Aim and Scope**

As students are going through their academics and pursuing their interested courses, it is very important for them to assess their capabilities and identify their interests so that they will get to know in which career area their interests and capabilities are going to put them in. This will help them in improving their performance and motivating their interests so that they will be directed towards their targeted career and get settled in that. Also recruiters while recruiting the candidates after assessing them in all different aspects, these kind of career recommender systems help them in deciding in which job role the candidate should be kept in based on his/her performance and other evaluation

## **2. Literature Survey**

### **2.1 Previous Work in the field**

Career goals, especially choosing the appropriate career through monitoring of the scope and trends in computer science and engineering job dimensions have been a prime need for all computer science undergraduate youngsters. It has always been essential for an early signal to help the developmental mindset. The papers have made an attempt to explore dynamic dataset and apply data mining based methods to explore student's insights based on characteristics related to academic, technical and interpersonal factors.

Artificial Neural Networks [1] is a data processing system consisting of a large number of simple, highly interconnected processing elements in an architecture inspired by the structure of the cerebral cortex portion of the brain. Hence, neural networks are often capable of doing things which humans or animals do well but which conventional computers often do poorly. For career recommendation various parameters (which are mentioned in section II. proposed system, subsection B. model) are considered which becomes quite difficult to predict using traditional regression models. In recent years, recommendation systems [2] have been widely used in various commercial platforms to provide recommendations for users. Every field has various job roles which makes it challenging for any undergraduate student and recruiter to decide a well-suited job for students. Any student after graduation needs to decide which job role is best suited for him according to his profile. This is important for a long-term career plan. Similarly, for a recruiter it is very crucial to recruit a candidate after assessing him/her in all different aspects. A job role prediction system will help undergraduate students and recruiters in finding the right job based on their personality, academics, interests, etc. Roshani & Deshmukh (2014) [7] have proposed an ensemble incremental learning algorithm created by using the set of three classifiers namely, Naive Bayes, K-Star and SVM. This was found to be a useful technique for offering the best career choice for the student. Furthermore, Arafath, et al. (2018)[14] their research helped predict student's estimated careers including student's strengths and weaknesses.

The above papers have used machine learning techniques such as SVM, Decision tree, Random forest, KNN, Naive Bayes algorithms. Amongst these algorithms SVM is proven to give the best accuracy. These algorithms have been trained on a small dataset with few class divisions. Features like gender, sector,

reservation, rank, student initial condition, age, educational level of parents, emotional factors, instructor difficulty, problem solving skill, professional skill, enthusiasm, final year project type, teamwork ability, communication skill, research background, strength, attitude, discipline, social ability, adaptability, leadership, team work, career.

In this research work, an attempt to explore a dynamic dataset and apply data mining based methods to explore student's insights based on characteristics related to academic, technical and interpersonal factors. This research helped predict a student's estimated career including student's strengths and weaknesses. The accuracy of prediction actually lies with the set of relevant skill parameters, interpersonal and academic factors. The research also helped teachers identify the students who need special attention and allowed the teacher to provide appropriate counselling as well as give them a proper guideline for selecting a specific job sector which leads to a healthy collaboration between academia and industry. The model was tested and found performing well in a constraint based learning environment. This paper discusses part of the main work in the field of data science, mining and analytics. Family of algorithms is developed to predict the educational relevance of individuals' talents through a lens of personality features (unstructured and semi-structured) and academic/career data. This paper presents progress of results in Good Fit Students (GFS) algorithms and math construct. This work addresses the problems of poor academic performances, low retention rates, drop outs, school transfers, costly readmissions, poor job performances, early job transfers and inefficient utilization/consideration of natural talents. GFS builds a framework and algorithms by correlating and blending social networking personality traits data with academic and career data. The results are promising at this stage of research and show improved predictions and relevant probabilities. Future work is focused on improving the results with more data and adding a few more algorithms to the main research/framework.

The awareness of choosing the right option for career is increasing among the students. The students fail to know their strengths and choose their career randomly which leads to frustration and demoralization. The automated systems used by the counsellors are to evaluate the personality traits of the individual. The accuracy of prediction depends on the set of relevant skill parameters and analytics on these. The research in view to design the accurate prediction and recommendation systems shows use of mining and statistical algorithms.

## **2.2 Existing System**

In research papers, models such as SVM, KNN, Naive Bayes, Decision tree and Random forest algorithm are used. There are no proper applications available to predict which job profile the person would be fit based on different criterias in the area of computer science and engineering.

## 3. Proposed System

### 3.1 Proposed Solution

We propose a job role recommendation system using neural networks due to the high number of parameters for classification. These parameters include student performance in various subjects present in the undergraduate curriculum of computer science as well as student interests, interpersonal skills, talents, etc. [6, 7, 8]. This project aims to implement the concept using an Artificial Neural Network(ANN) model. The model is trained and tested on 15,000 and 3,000 dataset entries respectively. The model performs multiclass classification and is able to predict one of the 6 domains (i.e. Database Administrator, Project Manager, Software Developer, Business Intelligence Analyst, Security Administrator, Technical Support).

### 3.2 Methodology Overview

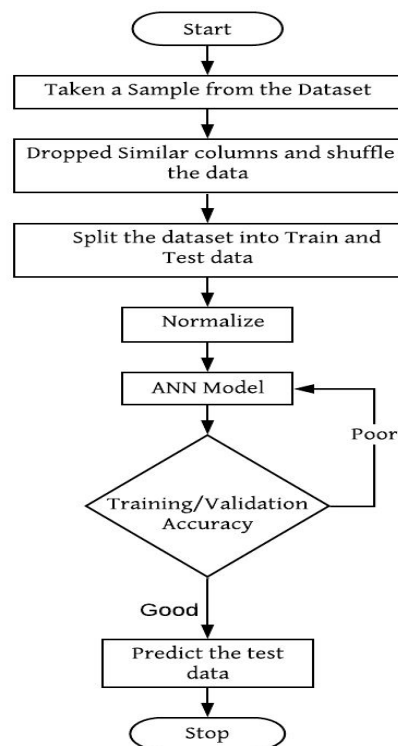


Fig 1: ANN Flowchart

- I. We took a small sample of data in a random manner which made sure that there were all the domains included in the sample dataset.

- II. The data was then cleaned by dropping out a few features such as Salary Range Expected, Interested Type of Books, talent tests, olympiads, Type of company wanting to settle in?, Interested subjects. Since they were repetitive and were not required for the prediction.
- III. The dataset was split into 70:20:10 ratio in training, testing and validation.
- IV. Suggested job role field was categorical data so OneHot encoding was performed to get into a binary form. There were parameters in the dataset which had different ranges, so Standard Scaler python library was used on the dataset to normalize the data.
- V. Our Artificial Neural Network model trained on the training data and changed its parameter according to the Validation and Testing dataset. This process continued till we acquired a global minima for the cost function.

### 3.3 Architecture

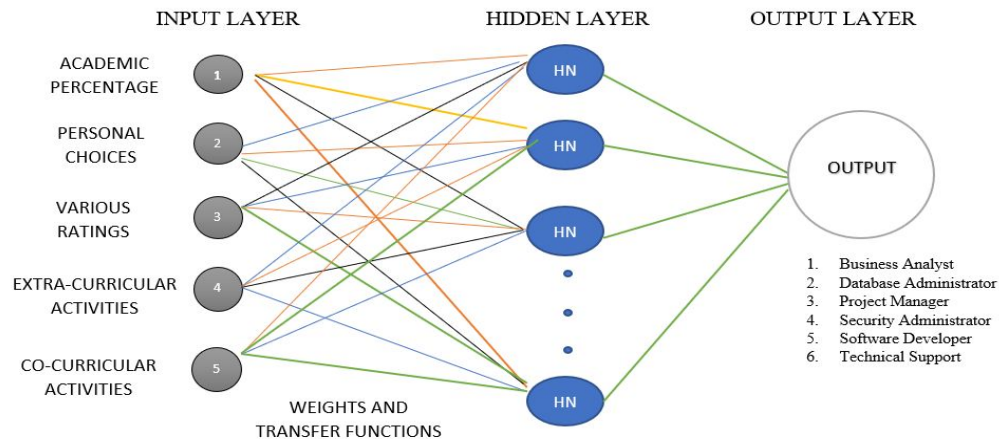


Fig 2: ANN Architecture

An artificial neuron network (ANN) is a computational model based on the structure and functions of biological neural networks. The proposed ANN model consists of an input layer, 10 hidden layers and an output layer which uses a Sequential model.

The input layer is given inputs such as academic percentages, extra & co-curricular activities, various activities and personal choices i.e 15 input features. There are 20 nodes in total which take input and using



activation function give the output to the next layer. Here, the Rectified linear unit(ReLU) function was used instead of sigmoid and tanh functions since it gave better accuracy and was less expensive operation than the other two. Further, L1 regularizer was used instead of L2 as it was shrinking the less important features coefficient to zero.

The 2nd layer consisted of 20 nodes which take input from the first layer. The ReLu function was used in this layer just like the first layer. The next 9 layers are the same as the 2nd layer leaving the fact that it receives the input from its previous layer.

The last layer consists of 6 outputs because the model predicts one of the 6 domains due to which the model uses multiclass classification in the last layer. Softmax activation function was used in the model for multiclass classification which gave fairly good results. To select a particular domain the model sets probability to each class so that it can select the class with maximum probability, it is done using a categorical cross entropy.

## 4. Implementation

### 4.1 Dataset

Our dataset consists of 18,000 entries from students with their marks, personal interest and extra & co-curricular and the job role they were suited for. For the model we considered 15 parameters. The dataset was split into 70:20:10 ratio in training, testing and validation. Original dataset was as follow:

A	B	C	D	E	F	G	H	I
Academic percentage in Operating Systems	percentage in Algorithms	Percentage in Programming	Percentage in Software Engineering	Percentage in Computer Network	Percentage in Electronics Sub	Percentage in Computer Architect	Percentage in Mathematics	Percentage in Communication skills
69	63	78	87	94	94	87	84	61
78	62	73	60	71	70	73	84	91
71	86	91	87	61	81	72	72	94
76	87	60	84	89	73	62	88	69
92	62	90	67	71	89	73	71	73
88	86	62	79	93	84	69	71	82
93	77	69	79	90	93	73	63	77
84	72	88	62	66	63	78	94	60
73	66	66	81	81	69	61	87	90
62	76	85	91	82	69	63	63	81

J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Hours working per	Logical quotient rating	hackathons	coding skills rating	public speaking point	can work long time before	self-learning capability	Extra-courses did	certifications	workshops	talent tests taken?	olympiads	reading and writing skills	memory capability score
9	4	0	4	8	yes	yes	yes	shell programming	cloud computing	no	yes	excellent	excellent
12	7	1	2	3	yes	no	yes	machine learning	database security	no	no	poor	medium
11	1	4	1	3	yes	no	yes	app development	web technologies	no	yes	poor	excellent
7	1	1	2	5	no	yes	no	python	data science	yes	no	medium	excellent
4	5	4	6	3	no	no	no	app development	cloud computing	no	no	poor	excellent
11	5	3	8	1	yes	yes	no	python	database security	no	no	poor	medium
6	3	2	3	3	no	no	yes	r programming	cloud computing	no	no	excellent	poor
12	2	1	6	6	yes	no	no	r programming	cloud computing	no	yes	poor	excellent
10	5	2	4	8	no	yes	no	information security	testing	yes	no	poor	poor
10	9	0	5	4	no	no	yes	app development	data science	yes	yes	excellent	poor

X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK
Interested subjects	interested career area	Job/Higher Studies?	Type of company wanted	Taken inputs from seniors	interested in games	Interested Type of Books	Salary Range Expected	In a Relationship?	Gentle or Tuff behavior	Management or Technical	Salary/work	hard/smart worker	worked in teams ever
cloud computing	system developer	higher studies	Web Services	no	no	Prayer books	salary	no	stubborn	Management	salary	hard worker	yes
networks	Business process analyst	job	SaaS services	yes	yes	Childrens	salary	yes	gentle	Technical	salary	hard worker	no
hacking	developer	higher studies	Sales and Marketing	yes	yes	Travel	Work	no	stubborn	Management	work	hard worker	no
networks	testing	higher studies	Testing and Maintenance	no	no	Romance	Work	yes	gentle	Management	work	smart worker	yes
Computer Architecture	testing	higher studies	product development	no	yes	Cookbooks	salary	no	stubborn	Management	work	hard worker	yes
programming	system developer	job	Testing and Maintenance	yes	yes	Self help	salary	no	stubborn	Management	work	smart worker	yes
parallel computing	security	job	BPA	no	no	Drama	Work	yes	gentle	Technical	salary	hard worker	yes
parallel computing	developer	higher studies	BPA	no	no	Romance	salary	no	stubborn	Technical	salary	smart worker	no
IOT	developer	higher studies	BPA	no	no	Math	Work	no	gentle	Technical	work	hard worker	yes
data engineering	Business process analyst	job	product development	no	no	Religion-Spirituality	Work	yes	gentle	Management	salary	hard worker	yes

AL	AM	AN
Introvert	Suggested Job Role	
no	Database Administrator	
yes	Project Manager	
yes	Project Manager	
yes	Security Administrator	
yes	Business Intelligence Analyst	
no	Software Developer	
yes	Database Administrator	
no	Business Intelligence Analyst	
yes	Business Intelligence Analyst	
no	Technical Support	
no	Software Developer	

#### 4.1.1 Input Parameters

No.	Name	Type
1.	Percentage <ul style="list-style-type: none"> <li>1. Operating Systems</li> <li>2. Algorithms</li> <li>3. Programming Concepts</li> <li>4. Software Engineering</li> <li>5. Computer Networks</li> <li>6. Mathematics</li> <li>7. Communication Skills</li> </ul>	Academics
2.	Hackathons	Co-Curricular
3.	Certifications <ul style="list-style-type: none"> <li>1. App Development</li> <li>2. FullStack Development</li> <li>3. Hadoop</li> <li>4. Information Security</li> <li>5. Machine Learning</li> <li>6. Python</li> </ul>	Co-Curricular

	7. Shell Programming	
4.	Interests <ol style="list-style-type: none"> <li>1. Analyst</li> <li>2. Management</li> <li>3. Programming</li> <li>4. Cloud Computing</li> <li>5. Data Engineer</li> <li>6. Software Engineering</li> <li>7. System Developer</li> <li>8. Hacking</li> <li>9. Networks</li> <li>10. Security</li> <li>11. Web Development</li> </ol>	Co-Curricular
5.	Team-spirit (sports)	Extra-Curricular
6.	Leadership Ability	Personality
7.	Introvert/ Extrovert	Personality
8.	Self Learning capability	Personality
9.	Management/Technical	Personality

#### 4.1.2 Output Parameters

No.	Job Profile
1	Business Analyst
2	Database Administrator
3	Project Manager

4	Security Administrator
5	Software Developer
6	Technical Support

## 4.2 Training Parameters

Parameter	Symbol	Value
Learning Rate	$\alpha$	0.0001
Exponential decay rate for moment	$\beta_1$	0.9
	$\beta_2$	0.999
Epsilon	$\epsilon$	$10^{-8}$
Epochs	-	1000
Minibatch Size	-	512

### 4.3 Code for Feature Extraction

```
import pandas as pd
import numpy as np
tr      = "../Data/train.csv"
ts      = "../Data/test.csv"
train = pd.read_csv(tr)
test  = pd.read_csv(ts)
#wines = np.genfromtxt("../rool.csv", delimiter=";", skip_header=1)
data = np.loadtxt(tr, delimiter=',', skiprows=1)
print(data.shape)
myu,sig = [],[]
for i in range(13):
    mean = np.mean(data[:,i])
    myu.append(mean)
    sigm = np.sum(np.square(data[:,i]-mean))/(data.shape[0]-1)
    sigm = sigm**0.5
    sig.append(sigm)
    data[:,i] = (data[:,i]-mean)/sigm

print(myu,sig)
# Normalising the marks columns (1-9)
# [77.00318789848731, 76.99831228903614, 77.07569696212026, 77.11301412676585, 76.9541817727216,

for i in range(13):
    train.iloc[:,i] = (train.iloc[:,i] - myu[i])/sig[i]
    test.iloc[:,i] = (test.iloc[:,i] - myu[i])/sig[i]

with open(tr, 'w', encoding='utf-8') as f:
    train.to_csv(f, index=False)

with open(ts, 'w', encoding='utf-8') as f:
    test.to_csv(f, index=False)
```

```

import pandas as pd

tr = "./train.csv"
ts = "./test.csv"
train = pd.read_csv(tr)
test = pd.read_csv(ts)

train.drop(['Hours working per day', 'public speaking points', 'can work long time before system?_no', 'can work long time before system?_y', 'workshops_hacking', 'workshops_system designing', 'workshops_testing', 'workshops_web technologies', 'talenttests taken?_no', 'talenttests', 'Type of company want to settle in?_BPA', 'Type of company want to settle in?_Cloud Services', 'Type of company want to settle in?_Finance', 'Type of company want to settle in?_Service Based', 'Type of company want to settle in?_Testing and Maintenance Services', 'Type of company', 'interested in games_no', 'interested in games_yes', 'In a Relationship?_no', 'In a Relationship?_yes', 'Gentle or Tuff behaviour?_gentle', 'Salary/work_salary', 'Salary/work_work', 'hard/smart worker_hard worker', 'hard/smart worker_smart worker', 'worked in teams ever?_no', 'wo', 'Introvert_no', 'Introvert_yes'], axis=1, inplace=True)

test.drop(['Hours working per day', 'public speaking points', 'can work long time before system?_no', 'can work long time before system?_y', 'workshops_hacking', 'workshops_system designing', 'workshops_testing', 'workshops_web technologies', 'talenttests taken?_no', 'talenttests', 'Type of company want to settle in?_BPA', 'Type of company want to settle in?_Cloud Services', 'Type of company want to settle in?_Finance', 'Type of company want to settle in?_Service Based', 'Type of company want to settle in?_Testing and Maintenance Services', 'Type of company', 'interested in games_no', 'interested in games_yes', 'In a Relationship?_no', 'In a Relationship?_yes', 'Gentle or Tuff behaviour?_gentle', 'Salary/work_salary', 'Salary/work_work', 'hard/smart worker_hard worker', 'hard/smart worker_smart worker', 'worked in teams ever?_no', 'wo', 'Introvert_no', 'Introvert_yes'], axis=1, inplace=True)

with open(tr, 'w', encoding='utf-8') as f:
    train.to_csv(f, index=False)

with open(ts, 'w', encoding='utf-8') as f:
    test.to_csv(f, index=False)

#43 inputs columns and 6 output columns available

```

```

import numpy as np #used to import mathematical operations
import matplotlib.pyplot as plt #used to plot different things in python
import pandas as pd #import data sets and manage data sets

#import dataset
#so make the directory so store dataset
dataset = pd.read_csv('para_data_train_shuffle.csv')
X_train = dataset.iloc[:, :38].values #independant variable vector
#iloc[] means first : is for rows(i.e.select all the lines) and 2nd : is for c
Y_train = dataset.iloc[:, 38:].values

dataset1 = pd.read_csv('para_data_test_shuffle.csv')
X_test = dataset1.iloc[:, :38].values #independant variable vector
#iloc[] means first : is for rows(i.e.select all the lines) and 2nd : is for c
Y_test = dataset1.iloc[:, 38:].values

from sklearn.preprocessing import StandardScaler
sc_X = StandardScaler()
X_train = sc_X.fit_transform(X_train)
X_test = sc_X.transform(X_test)

```



## 4.4 Code for ANN

```
import numpy as np
import pandas as pd

from sklearn.preprocessing import StandardScaler
from sklearn.decomposition import PCA

from keras.models import Sequential
from keras.utils import np_utils
from keras.layers import Dense, Dropout, GaussianNoise, Conv1D
from keras.preprocessing.image import ImageDataGenerator
from keras import regularizers

import matplotlib.pyplot as plt
import seaborn as sns
model = Sequential()
layers = 10
units = 15

model.add(Dense(units, input_dim=38, activation='relu', kernel_regularizer=regularizers.l1(0.1)))
#model.add(GaussianNoise(pca_std))
for i in range(layers):
    model.add(Dense(units, activation='relu'))
    #model.add(GaussianNoise(pca_std))
    #model.add(Dropout(0.1))
model.add(Dense(6, activation='softmax'))

model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['categorical_accuracy'])

history = model.fit(X_train, Y_train, epochs=1000, batch_size=512, validation_split=0.30, verbose=2)

_, test_acc = model.evaluate(X_test, Y_test, verbose=1)
print('Test: ', test_acc)

from keras.models import load_model

model.save('Job_Role_model_v2.h5')
print("Model saved")

import pickle
pickle.dump(model, open('Job_Role_Pickle.sav', 'wb'))
print("Model saved as pickle")
```



```

loss = history.history['loss']
val_loss = history.history['val_loss']
epochs = range(1, len(loss) + 1)
plt.plot(epochs, loss, 'y', label='Training loss')
plt.plot(epochs, val_loss, 'r', label='Validation loss')
plt.title('Training and validation loss')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.legend()
plt.show()

print(history.history())

acc = history.history['acc']
val_acc = history.history['val_acc']
plt.plot(epochs, acc, 'y', label='Training acc')
plt.plot(epochs, val_acc, 'r', label='Validation acc')
plt.title('Training and validation accuracy')
plt.xlabel('Epochs')
plt.ylabel('Accuracy')
plt.legend()
plt.show()

```

## 4.5 Code for Flask

```

@app.route('/prediction',methods=['POST'])
def prediction():

    os      = request.form["os"]
    aoa     = request.form["aoa"]
    pc      = request.form["pc"]
    se      = request.form["se"]
    cn      = request.form["cn"]
    ma      = request.form["ma"]
    cs      = request.form["cs"]
    hac     = request.form["hac"]
    interest = request.form["interest"]
    cert    = request.form["cert"]
    personality = request.form["personality"]
    mantech = request.form["mantech"]
    leadership = request.form["leadership"]
    team    = request.form["team"]
    selfab  = request.form["selfab"]

    myu = [77.00318789848731, 76.99831228903614, 77.07569696212026, 77.11301412676585, 76.9541817727216, 77.0150018752344, 77.060320040005, 5.
    sig = [10.071578660726848, 10.098653693844197, 10.137528173238477, 10.088164425588161, 10.018397202418788, 10.18533143324003, 10.095941558
    arr = [os,aoa,pc,se,cn,ma,cs,hac]

    for i in range(8):
        arr[i] = float(arr[i])
        arr[i] = (arr[i]- myu[i])/sig[i]

    inti    = [0,0,0,0,0,0,0,0,0,0,0]
    certi   = [0,0,0,0,0,0,0]

```

```

if interest == "analyst":
    inti[0] = 1
elif interest == "hadoop":
    inti[1] = 1
elif interest == "cloud":
    inti[2] = 1
elif interest == "data":
    inti[3] = 1
elif interest == "hacking":
    inti[4] = 1
elif interest == "management":
    inti[5] = 1
elif interest == "networks":
    inti[6] = 1
elif interest == "programming":
    inti[7] = 1
elif interest == "security":
    inti[8] = 1
elif interest == "software":
    inti[9] = 1
elif interest == "system":
    inti[10] = 1
elif interest == "testing":
    inti[11] = 1
elif interest == "web":
    inti[12] = 1

if cert == "app":
    certi[0] = 1
elif cert == "full":
    certi[1] = 1
elif cert == "hadoop":
    certi[2] = 1
elif cert == "security":
    certi[3] = 1
elif cert == "machine":

with graph.as_default():
    print("done in loop")
    y = model.predict(np.array([arr,]))
    result = np.where(y == np.amax(y))
    print(result[0])

print("done1")

if result[1]==[0]:
    print('Business Intelligence Analyst')
    return render_template('jobprofile.html', prediction_text='Business Intelligence')

elif result[1]==[1]:
    print('Database Administrator')
    return render_template('jobprofile.html', prediction_text='Database Administrator')

elif result[1]==[2]:
    print('Project Manager')
    return render_template('jobprofile.html', prediction_text='Project Manager')

elif result[1]==[3]:
    print('Security Administrator')
    return render_template('jobprofile.html', prediction_text='Security Administrator')

elif result[1]==[4]:
    print('Software Developer')
    return render_template('jobprofile.html', prediction_text='Software Developer')

else:
    print('Technical Support')
    return render_template('jobprofile.html', prediction_text='Technical Support')

print("done2")

```

## **5. Technologies Used**

### **5.1 Python**

Python is a powerful high-level, object-oriented programming language created by Guido van Rossum. It has simple easy-to-use syntax, making it the perfect language for someone trying to learn computer programming for the first time. Python is a general-purpose language. It has a wide range of applications from Web development (like: Django and Bottle), scientific and mathematical computing (Orange, SymPy, NumPy) to desktop graphical user Interfaces (Pygame, Panda3D). The syntax of the language is clean and the length of the code is relatively short. It's fun to work in Python because it allows you to think about the problem rather than focusing on the syntax.

### **5.2 Visual Studio**

Visual Studio Code is a lightweight but powerful source code editor which runs on your desktop and is available for Windows, macOS and Linux. It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages (such as C++, C#, Java, Python, PHP, Go) and runtimes (such as .NET and Unity).

### **5.3 Libraries used**

#### **5.3.1 Tensorflow**

TensorFlow is an end-to-end open source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML powered applications.

#### **5.3.2 Flask**

Flask is a lightweight WSGI web application framework. It is designed to make getting started quick and easy, with the ability to scale up to complex applications. It began as a simple wrapper around Werkzeug and Jinja and has become one of the most popular Python web application frameworks

### **5.3.3 Scikit-learn**

Scikit-learn (formerly scikits.learn) is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy. Clustering of unlabeled data can be performed with the module `sklearn.cluster`. From `sklearn.cluster` we can import `KMean` which is used in our application.

### **5.3.4 Keras**

Keras was created to be user friendly, modular, easy to extend, and to work with Python. The API was “designed for human beings, not machines,” and “follows best practices for reducing cognitive load.” Neural layers, cost functions, optimizers, initialization schemes, activation functions, and regularization schemes are all standalone modules that you can combine to create new models. New modules are simple to add, as new classes and functions. Models are defined in Python code, not separate model configuration files.

### **5.3.5 Jinja**

Jinja is a modern and designer-friendly templating language for Python, modelled after Django’s templates. It is fast, widely used and secure with the optional sandboxed template execution environment

### **5.3.6 SQLAlchemy**

SQLAlchemy is the Python SQL toolkit and Object Relational Mapper that gives application developers the full power and flexibility of SQL. It provides a full suite of well known enterprise-level persistence patterns, designed for efficient and high-performing database access, adapted into a simple and Pythonic domain language.

## 6 Hardware and Software Requirements

### The Hardware Requirements

Processor	Intel i5 or i7
RAM	Minimum 4GB
GPU	NVIDIA or Intel HD Graphics
Memory	20GB
Operating System	Windows 10 or macOS Catalina 10.15.4 or Ubuntu 18.04

### The Software Requirements

Software	Version
Flask	1.1.1
SQLAlchemy	1.2
Python	3.5.3
Keras	2.2.4
Tensorflow	1.5

## 7. Project Timeline

Week No.	Work Done
July 3rd & 4th Week	We selected our Project topic and started exploring it by reading various research papers related to the topic.
August 1st & 2nd Week	We found out the deep learning technologies need for our topic and starting learning it by doing various online courses and tutorial
August 3rd & 4th Week	Data Gathering & Data Cleaning is the most important step of any deep learning project, besides this we also started implementing different deep learning algorithms.
September 1st & 2nd Week	We started implementing different feature extraction techniques such as DropOut and Normal
September 3rd & 4th Week	We started to train our models using SVM and Decision Tree. We initially approached these two models due to less accuracy from these two models, We thought with the approach of training our model using ANN.
October 1st Week	Second Internal Mid-Term Presentation
October 2nd, 3rd & 4th Week	Due to submissions , oral & practical exams , not much progress was made during this period. Based on the feedback of Internal Presentation , we tuned our project to present a better demo.
November 1st Week	Mid-Term Presentation

December 3rd & 4th Week	After our Semester Exams, we brainstormed on feedback given by the external examiner. We also started implementing our system with increased dataset & features.
January 1st & 2nd Week	We freezed our ANN model which gave us a good accuracy of 94.9%. We started writing technical papers on the project.
February 1st , 2nd & 3rd Week	We dedicated 3 weeks to finalize our research paper. We trained and tested our system against a large dataset and added results to our research paper.
February 4th Week	Our paper got accepted and we presented our research paper at IJCTT-2020
March 1st & 2nd Week	We started working on the website design, the technologies used are HTML, CSS, Bootstrap, SQL, AngularJS and Flask.
March 3rd & 4th Week	We deployed our ANN model with our Website. We started preparing our blackbook
April 1st	Final testing of the system was done, and blackbook was completed

## 8. Conclusion and Future Works

### 8.1 Result Analysis

The proposed Career Recommender System is being trained on 18,000 datasets having 15 parameters which gives an accuracy of 99% whereas testing on 3,000 datasets with same parameters gave an accuracy of 94.9%. In comparison with Roy, K. et al. (2018) [10] model who used traditional machine learning algorithms like SVM, XGBoost and Decision our model proved to give better accuracy than theirs. The cost function shows an exponential decrease as follows:

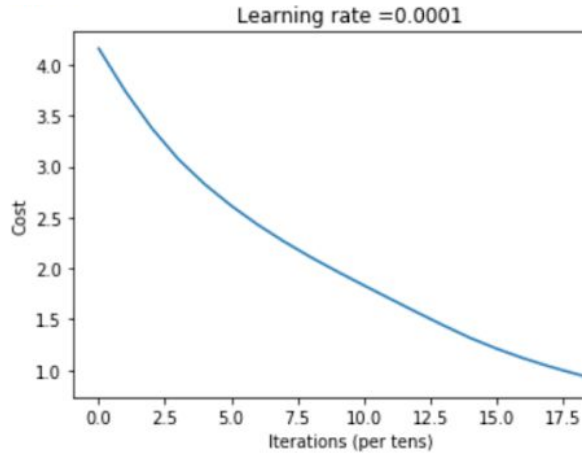


Fig 4: Cost Function

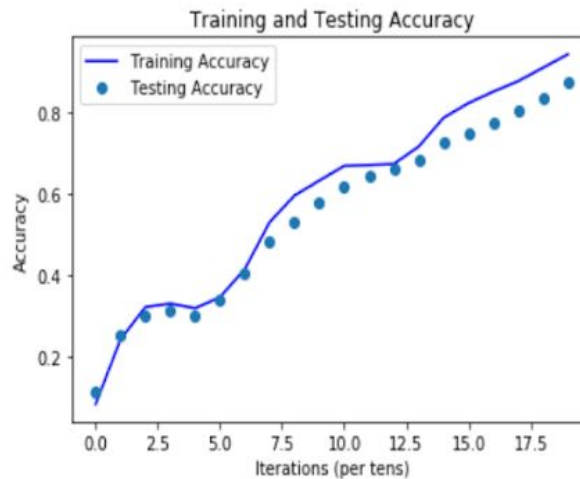


Fig 5: Training and Testing Accuracy

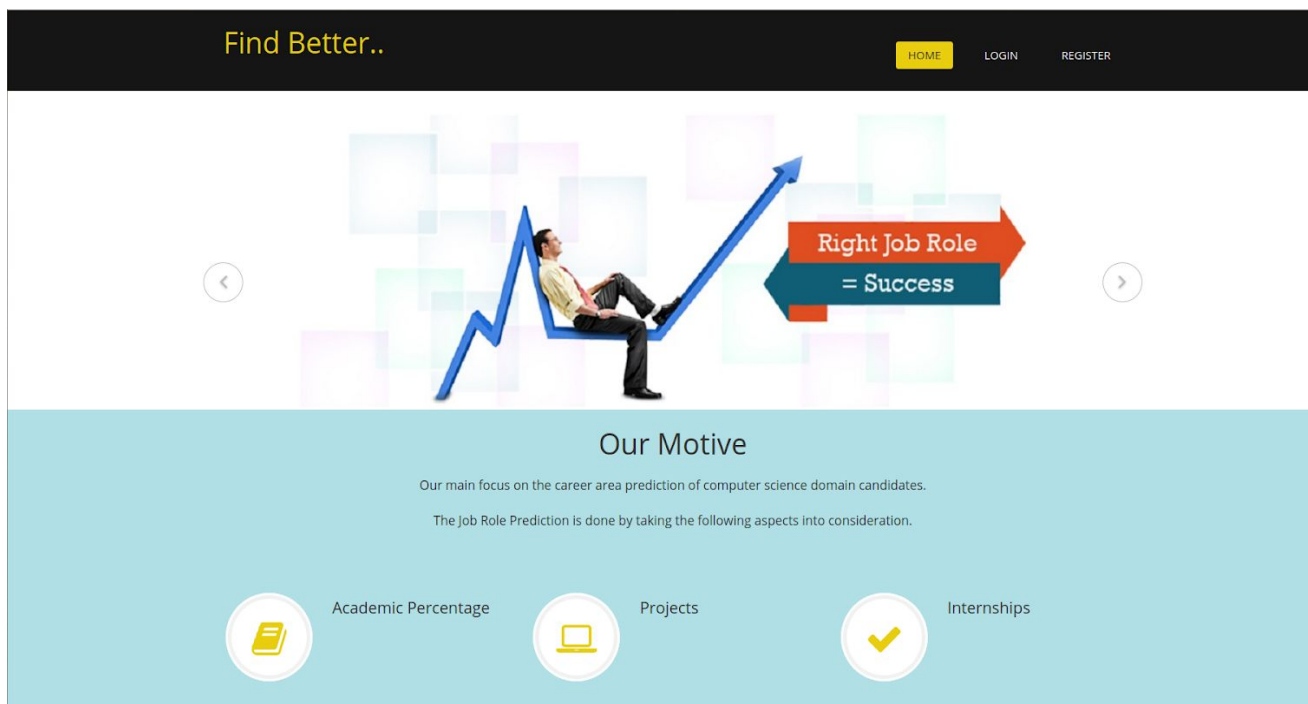


## 8.2 Conclusion

The developed model is apt for the analysis of many objective factors for a person with qualified knowledge, and skills. This recommender system can be used by any IT based recruiter to hire a candidate appropriate for the job. Additionally, an individual as a Computer Engineering fresher can find out the domain that they are qualified for based on their profile and the ones who are unaware of their career. The proposed model used 15 parameters to predict one of the six job-roles with an accuracy of 94.9%. Hence, ANN models give more accurate results to traditional machine learning models.

## 8.3 Snapshots

Here we sequentially show the SNAPSHOT'S of the working model of our project.



Which Job-Profile best suits You??



Which Skills required are for?



## CREATE ACCOUNT

First Name

juhi

Last Name

checker

College

crce

Email

juhi@gmail.com

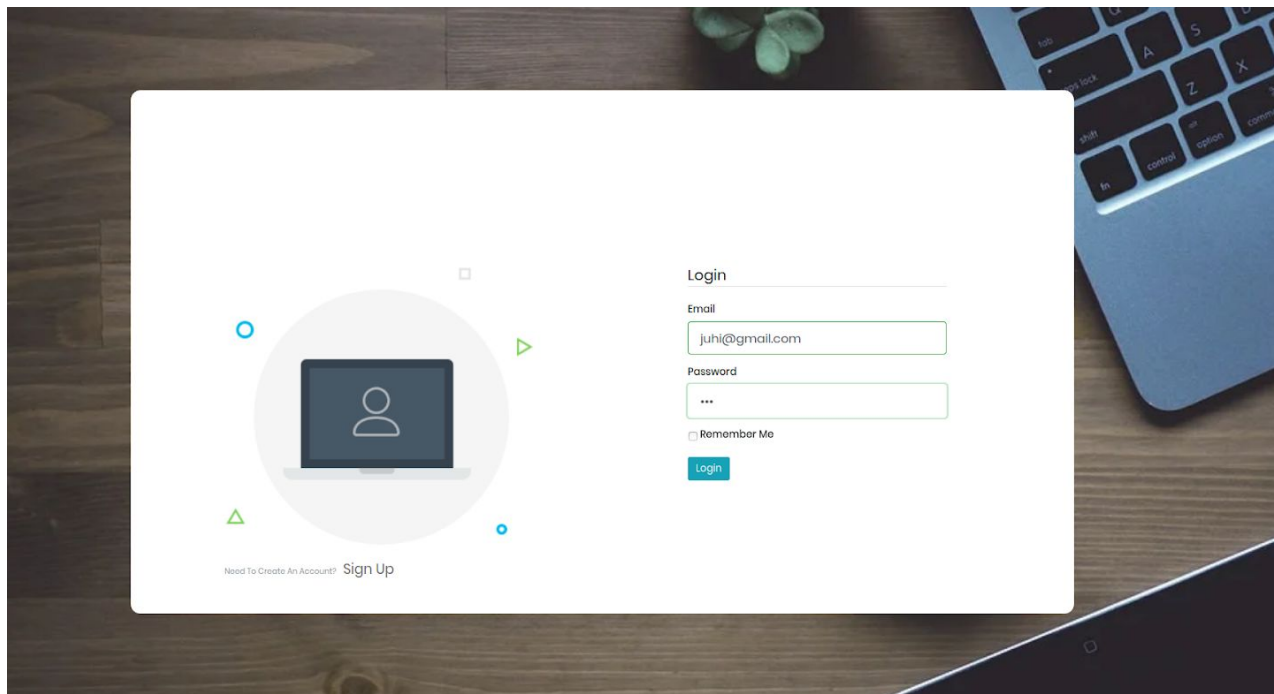
Password

\*\*\*

Confirm Password

\*\*\*

REGISTER

Already Have An Account? [Sign In](#)

## Which Job-Profile best suits You??

Operating System	85
Analysis of Algorithm	75
Programming Concept	65
Software Engineering	72
Computer Network	86
Applied Mathematics	70
Computer Security	73
Hackathons attended	2
Interest	Software Engineer

Applied Mathematics	70
Computer Security	73
Hackathons attended	2
Interest	Software Engineer
Topmost Certification	Full Stack
Personality	<input checked="" type="radio"/> Introvert <input type="radio"/> Extrovert
Management or Technical	<input checked="" type="radio"/> Management <input type="radio"/> Technical
Leadership	<input checked="" type="radio"/> Yes <input type="radio"/> No
Team	<input checked="" type="radio"/> Yes <input type="radio"/> No
Self Ability	<input checked="" type="radio"/> Yes <input type="radio"/> No

Predict

Find Better..

HOME

ACCOUNT

SKILLS

RESUME BUILDER

LOGOUT

Hackathons attended

2

Interest

Software Engineer

Topmost Certification

Full Stack

Personality

☒ Introvert ☐ Extrovert

Management or Technical

☐ Management ☒ Technical

Leadership

☒ Yes ☐ No

Team

☒ Yes ☐ No

Self Ability

☒ Yes ☐ No

Predict

68%

Waiting for 127.0.0.1...

Find Better..

HOME

ACCOUNT

SKILLS

RESUME BUILDER

LOGOUT

Leadership

☒ Yes ☐ No

Team

☒ Yes ☐ No

Self Ability

☒ Yes ☐ No

Predict

100%

Technical Support 😊



Leadership ☒ Yes ☐ No

Team ☒ Yes ☐ No

Self Ability ☒ Yes ☐ No

Predict

100%

Software Developer 😊



© AJS. All Rights Reserved.

Home Faq Contact Us

Learn it here..


Home

Categories >

Aptitude >


CODING

- Hackerrank
- Codechef
- Geeks for Geeks
- Hacker Earth
- Free Code Camp



Videos


Sort by ▾



C Programming Tutorials

Programming 🟢


1.8M views 11 Months ago



C++ Programming Tutorials

Programming 🟢


1.8M views 11 Months ago



Java Programming Tutorials

Programming 🟢

1.8M views 11 Months ago



Python

Education 🟢

1.8M views 11 Months ago



## 9. References

- [1] R. E. Uhrig, "Introduction to artificial neural networks," *Proceedings of IECON '95 - 21st Annual Conference on IEEE Industrial Electronics*, Orlando, FL, USA, 1995, pp. 33-37 vol.1.
- [2] S. Elayidom, S. M. Idikkula, J. Alexander and A. Ojha, "Applying Data Mining Techniques for Placement Chance Prediction," *2009 International Conference on Advances in Computing, Control, and Telecommunication Technologies*, Trivandrum, Kerala, 2009, pp. 669-671.
- [3] Bao, Xuecai & Wang, Tangsheng & Hu, Jianfeng. (2009). Analysis of Individual Career Decision-Making Based on Partial Least-Squares Regression Model. *6th International Conference on Fuzzy Systems and Knowledge Discovery*, FSKD 2009. 2. 522-526. 10.1109/FSKD.2009.192.
- [4] X. Bao, T. Wang and J. Hu, "Analysis of Individual Career Decision-Making Based on Partial Least-Squares Regression Model," *2009 Sixth International Conference on Fuzzy Systems and Knowledge Discovery*, Tianjin, 2009, pp. 522-526.
- [5] Guo, Bo & Zhang, Rui & Xu, Guang & Shi, Chuangming & Yang, Li. (2015). Predicting Students Performance in Educational Data Mining. 125-128. 10.1109/ISSET.2015.33.
- [6] A. Slim, G. L. Heileman, J. Kozlick and C. T. Abdallah, "Employing Markov Networks on Curriculum Graphs to Predict Student Performance," *2014 13th International Conference on Machine Learning and Applications*, Detroit, MI, 2014, pp. 415-418
- [7] Ade, Roshani & Deshmukh, P.R. (2014). An incremental ensemble of classifiers as a technique for prediction of student's career choice. *1st International Conference on Networks and Soft Computing*, ICNSC 2014 - Proceedings. 2014. 384-387. 10.1109/CNSC.2014.6906655.
- [8] L. S. Katore, B. S. Ratnaparkhi and J. S. Umale, "Novel professional career prediction and recommendation method for individuals through analytics on personal traits using C4.5 algorithm," *2015 Global Conference on Communication Technologies (GCCT)*, Thuckalay, 2015, pp. 503-506.
- [9] I. Naim, M. I. Tanveer, D. Gildea and M. E. Hoque, "Automated prediction and analysis of job interview



performance: The role of what you say and how you say it," *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, Ljubljana, 2015, pp. 1-6.

[10] Uddin, Muhammad & Lee, Jeongkyu. (2016). Predicting good fit students by correlating relevant personality traits with academic/career data. 968-975. 10.1109/ASONAM.2016.7752357

[11] H. Mallafi and D. H. Widyantoro, "*Prediction modelling in career management*," 2016 International Conference on Computational Intelligence and Cybernetics, Makassar, 2016, pp. 17-21.

[12] J.R. H. Rangnekar, K. P. Suratwala, S. Krishna and S. Dhage, "Career Prediction Model Using Data Mining and Linear Classification," *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, Pune, India, 2018, pp. 1-6.

[13] Roy, K. & Roopkanth, K. & Teja, V. & Bhavana, V. & Priyanka, J. (2018). Student Career Prediction Using Advanced Machine Learning Techniques. *International Journal of Engineering and Technology (UAE)*. 7. 26-29. 10.14419/ijet. v7i2.20.11738.

[14] Arafath, Yeasin & Saifuzzaman, Mohd & Ahmed, Sumaiya & Hossain, Syed. (2018). Predicting Career Using Data Mining. 10.1109/GUCON.2018.8674995.

[15] L. Zhang, T. Luo, F. Zhang and Y. Wu, "A Recommendation Model Based on Deep Neural Network," in *IEEE Access*, vol. 6, pp. 9454-9463, 2018.