

Implementation of Machine Learning Algorithms for Employee Recommendation

^{1st} Prof. Jyotsna More
B.E in Information Technology
Engineering Xavier Institute of
Engineering (University of Mumbai)

^{2nd} Bhavin Nirmal
B.E in Information Technology
Engineering Xavier Institute of
Engineering (University of Mumbai)

^{3rd} Namrata Bomble
B.E in Information Technology
Engineering Xavier Institute of
Engineering (University of Mumbai)

^{4th} Sayali Pawar
B.E in Information Technology
Engineering Xavier Institute of
Engineering (University of Mumbai)

Abstract— Generally, in organizations employees are hired based on their skill set and experiences which is done manually by collecting and analyzing their resumes. This process requires significant time and effort. To overcome this problem Employee Recommendation System will automate the process of selecting a set of employees from large number of applicants. This Machine learning model will be using a lot of resumes' data as a training dataset. System will analyze resumes and also recommend on the basis of certain parameters which are usually checked during the recruitment process. The system will take the candidate requirements from the organization in terms of the skill set, education and work experience required for the job. Then system will collect the resumes from applicants through a web interface and using natural language processing system will extract the required information necessary for prediction from resumes. Using a machine learning algorithm system will predict whether a particular candidate should be selected or not. The format of the resumes can be generic which gives users flexibility to upload the resumes in either .DOCX or .PDF format.

Keywords—Text mining, text extraction, natural language processing, recommendation.

I. INTRODUCTION

Candidates apply for a job post through various available portals such as company's website, external websites, job advertisements, job referrals, etc. After the candidates apply for the job, the recruitment phase of a company starts. Employee Recommendation System which is proposed is used in the selection of candidates from college campuses and other recruitment process. Job applicants basically respond to recruitment team and then screened to determine if they are eligible for a particular designation. This is a tedious job which involves lot of human interaction in the recruitment process. It might also involve various time consuming tasks like manually contacting candidates, verifying the resumes and so on. The recruitment team also needs to verify the background of the candidates for any unusual activities performed by him/her such as criminal records.

II. LITERATURE SURVEY

Hiring agencies require candidates to upload their resumes on agency's websites in particular formats. Agencies will go through structured data and preselected candidates for the company through curriculum. This process had a major drawback. There were multiple agencies and each had its own unique format. To address these problems, an intelligent algorithm is needed that will analyze the information of any

unstructured curriculum vitae, sort them by groups and classify them long last. Using Natural Language Processing and Machine Learning, the Employee Recommendation System will provide an efficient way to select candidates for a particular designation in a company [3]. This process will save a lot of time of Hiring Companies of manually selecting candidates by going through a stack of resumes that not only costs more but also occupies space. This is where an automated system comes into the picture and saves lot of human effort. It is necessary to preprocess the uploaded resumes and store the extracted information into one tuple for further processing [2].

III. BLOCK DIAGRAM

When the employee recommendation system is implemented in a company, the company needs to register itself in order to use the model. When the company starts the software, the first screen that will be popped on the machine will be the register screen. After the company registers on the software, the company has to collect resumes in a generic format which will be used as a data set for predicting the set of employees. The dataset will be in a tabular format which will be given to the software. The resumes will be collected from a web interface of the company which may have a different format and the resumes will be uploaded and saved in the back of the software. The data required for the prediction of candidates will be extracted from the resumes using text extraction and will be saved in a particular format [4]. The text mining of the resumes will be implemented using python. The companies will then have to set the parameters as the selection criteria to apply the algorithm and output the list of candidates eligible for a particular designation in a company. The algorithm will be implemented based on the accuracy of the algorithms being generated on the training dataset. The parameters in this model include the skills that are required for a particular job and the work experience the candidate has. Based on these parameters, the algorithm will be applied on the dataset and it will produce output for those values. Naïve Bayes algorithm is used to generate the output. The machine learning algorithm will be implemented using python to improve the performance of the system and generate the list of employees who are eligible for a particular post [8].

This automated learning model eventually improves the response time and performance of the system. The machine

learning model will be used to predict and recommend a suitable employee based on his/her skills which are extracted from his/her resume using Text Extraction techniques [2]. The Employee Recommendation System will also include a model for notifying the candidate about the selection process.

IV. DATASET DESCRIPTION

The dataset used for calculating the accuracy of the machine learning algorithms consists of the skills that will be used to further predict the model. The columns of the dataset include Internship Details, Academic Performance, Programming Skills, Project, Management Skills, and other relevant information. The values of the columns are numeric where a candidate is rated on a scale of 0 to 100. A snapshot of the dataset is given in Table I. The dataset also contains a column for placed and unplaced candidates for prediction purpose [1].

V. ALGORITHMS

A. NAÏVE BAYES ALGORITHM

The Bayesian Classification denotes a supervised learning method as well as a statistical method for classification. It takes an underlying probabilistic model and allows us to capture uncertainty about model fairly by determining probabilities of the results. The Bayesian Classification can solve diagnostic and probabilistic problems. Bayesian Classification offers practical learning algorithm and preliminary knowledge consequently [6]. Appropriate observed data can be combined. The Bayesian Classification provides a convenience approach to understand and evaluate many learning algorithms. It calculates explicit hints probabilities and is robust for noise in input data. Bayesian Classification is a simple classification algorithm which is useful in many applications [1]. A minimum of one author is required for all conference articles. Author names should be listed starting from left to right and then moving down to the next line. This is the author sequence that will be used in future citations and by indexing services. Names should not be listed in columns nor group by affiliation. Please keep your affiliations as succinct as possible (for example, do not differentiate among departments of the same organization).

B. LOGISTIC REGRESSION

In statistics, Logistic Regression (or logit) model is used to predict the probability of a certain event occurring such as the pass or fail, win or lose, alive or dead or healthy or sick. Basically Logistic Regression generates output considering only two possibilities which can be true/false, yes/no, etc. This can be extended to model many categories of events like deciding whether or not a picture contains a cat, dog or lion, etc. Each object which is detected within the image probability between 0 and 1 and the sum would be assigned by adding to 1. Logistic Regression is a statistical model that in its basic form uses a logistic function to model a dependent binary variable, although many more complex extensions [1].

C. KNN ALGORITHM

KNN is used for both classification and predictive regression. However, it is more widely employed in classification problems within the industry [7]. To evaluate any technique we generally observe 3 necessary aspects:

1. Ease to interpret the output
2. Calculation time

3. Predictive Power

Making Predictions with KNN:

KNN make predictions using the training dataset directly [5]. Predictions are made for a new instance (x) for those instances of K [1].

VI. FIGURES AND TABLES

The figures and table associated with the algorithms are shown below:

Table I. Dataset Description

A	B	C	D	E	F	G	H	I	J
Coding Sk	Technical	Core Know	Academic	Program	Managem	Projects	Internship	Backlog	Placed
85	90	40	75	90	40	80	80	0	1
89	85	50	70	80	40	70	75	0	1
80	90	55	78	88	45	85	80	0	1
95	90	40	60	80	30	70	0	0	0
95	80	80	88	85	70	90	80	0	1
90	80	85	76	83	79	79	80	0	1
70	80	60	78	70	60	60	70	0	1
50	70	65	74	65	70	70	66	0	1
60	74	70	80	70	74	65	72	0	1
90	85	65	79	90	83	75	79	0	1
70	69	73	79	72	86	70	81	0	1

TABLE I indicates the dataset used for calculating the accuracy of the algorithms used for Employee Recommendation System. There are various parameters like skills, experience for further process.

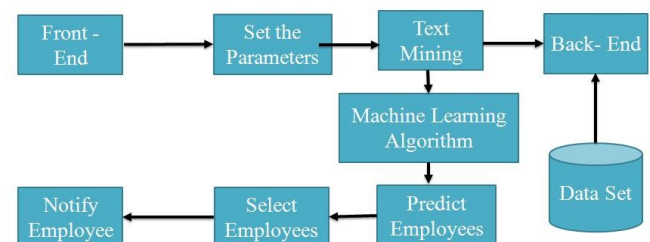


Fig. 1 BLOCK DIAGRAM

Fig. 1 indicates the flow of Employee Recommendation System through which the company can get results of the selected candidates who will be uploading resumes on the webpage.

```

from sklearn.naive_bayes import GaussianNB

#Create a Gaussian Classifier
gnb = GaussianNB()

#Train the model using the training sets
gnb.fit(x_train, y_train)

#Predict the response for test dataset
y_pred = gnb.predict(x_test)

from sklearn import metrics

# Model Accuracy, how often is the classifier correct?
print("Accuracy:",metrics.accuracy_score(y_test, y_pred))

Accuracy: 0.84
  
```

Fig. 2 NAÏVE BAYES RESULT

Fig. 2 shows the result obtained by training the dataset mentioned above using the Naïve Bayes Model. The accuracy obtained is 0.84% for Naïve Bayes Algorithm [6].

```

from sklearn.linear_model import LogisticRegression

lr = LogisticRegression(fit_intercept=True)
lr.fit(x_train,x_test)

LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
intercept_scaling=1, max_iter=100, multi_class='ovr', n_jobs=1,
penalty='l2', random_state=None, solver='liblinear', tol=0.0001,
verbose=0, warm_start=False)

predict=lr.predict(y_train)

print(metrics.accuracy_score(y_test,predict))

0.84

```

Fig. 3 LOGISTIC REGRESSION RESULT

Fig. 3 shows the result obtained by training the dataset mentioned above using the Logistic Regression Model. The accuracy obtained is 0.84% for Logistic Regression Algorithm.

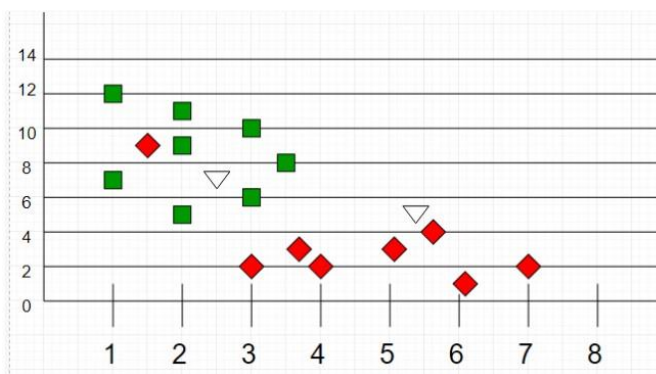


Fig. 4 KNN EXAMPLE

Fig. 4 shows the data points containing two features. Now, another set of data points are given which is also called testing data, assign these points to a group by analyzing the training set. Unclassified points are marked as 'White'. From the given unclassified point, one can assign that to a group by observing which group the nearest neighbors belong to. This means that a point near a group of points classifies as "Red" has a higher probability to be classified as "Red". In Employee Recommendation System the KNN model will be trained using the dataset consisting of skills required by the organization as a criteria to select the candidate, then given a new candidate's skill set (extracted from his/her resume using Text Extraction techniques) the KNN model will classify it as either a suitable or a non-suitable candidate [1].

```

#Import knearest neighbors Classifier model
from sklearn.neighbors import KNeighborsClassifier

#Create KNN Classifier
knn = KNeighborsClassifier(n_neighbors=5)

#Train the model using the training sets
knn.fit(x_train, y_train)

KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
metric_params=None, n_jobs=1, n_neighbors=5, p=2,
weights='uniform')

#Predict the response for test dataset
y_pred = knn.predict(x_test)

#Import scikit-learn metrics module for accuracy calculation
from sklearn import metrics
# Model Accuracy, how often is the classifier correct?
print("Accuracy:", metrics.accuracy_score(y_test, y_pred))

Accuracy: 0.88

```

Fig 5. KNN RESULT

Fig. 5 shows the result obtained by calculating the probabilities of the dataset using the KNN model [5].

VII. CONCLUSION

The system will take the candidate requirements from the organization in terms of the skill set, education and work experience required for the job. The next step will be collection of the resumes from applicants through a web interface and using natural language processing and will extract the required information necessary for prediction from resumes. Using a machine learning algorithm system will predict whether a particular candidate should be selected or not. The system will store all the applicants' information within the database to facilitate faster processing and future requirements. The general process of the Employee Recommendation System reduces the manpower in an industry by automating the recruitment process. This saves company.

REFERENCES

- [1] Anand Nautiyal, Satya Prakash Sahu, Mahendra Prasad, "Machine Learning Algorithms for Recommender System - a comparative analysis", 'International Journal of Computer Applications Technology and Research', 2017.
- [2] Pravin Shinde Sharvari Govilkar, "A Systematic study of Text Mining Techniques", 2015.
- [3] Satyaki Sanyal, Souvik Hazra, Soumyashree Adhikary, Neelanjana Ghosh, "Resume Parser with Natural Language Processing", 2017.
- [4] Sayed Zainul Abideen Mohd Sadiq, Juneja Afzal Ayub, Gunduka Rakesh Narsayya, Momin Adnan Ayyas, "Intelligent Hiring with Resume Parser and Ranking using Natural Language Processing and Machine Learning", 2016.
- [5] "KNN Classification using Scikit-learn," DataCamp Community. [Online]. Available: <https://www.datacamp.com/community/tutorials/knearest-neighbor-classification-scikit-learn>.
- [6] "Naive Bayes Classification using Scikit-learn," DataCamp Community. [Online]. Available: <https://www.datacamp.com/community/tutorials/naive-bayes-scikit-learn>.
- [7] Yun-lei Cai, Duo Ji, Dong-feng Cai, "A KNN research paper classification method based on Shared Nearest Neighbor", 2010.
- [8] Divyanshu Chandola, Aditya Garg, Ankit Maurya, Amit Kushwaha, "Online Resume Parsing System Using Text Analytics", 2015.