A USER PROFILE BASED RECOMMENDER SYSTEM FOR PRIVATE JOB MATCHING

Gajjela Monika, M. Sridevi
Associate Professor, Computer Science & Engineering, Anurag University, TS, India
Student, Computer Science & Engineering, Anurag University, TS, India

ABSTRACT

A job classification system is used to group positions that have similar duties and responsibilities. If done correctly, it creates parity in job titles and consistency in job levels within the organization hierarchy. Job classification is based upon various aspects of the job and does not take into consideration the person assigned. Instead, identified job value factors such as technologies, skills, qualifications, etc. are taken into consideration. These job value factors allow an organization to compare jobs which may not appear to be similar. Job value factors work because almost every job has them. From the dataset built by studying various LinkedIn profiles of the users, proposed methodology want to predict someone's job category based on his job summary using classification algorithms. Job summaries are created by users to describe their skills and tasks. Our goal is to use extract information from these free-form text fields and predict the occupation of the user. For this used Naïve Bayes and Support Vector Machine (SVM) algorithm. In this project, developed prototype application is described the working procedure of Job Prediction and shown performance results and prediction results using Python Django framework.

Keyword: - Naive Bayes, SVM, Job Prediction Performance

DOI Number: 10.48047/NQ.2022.20.20.NQ109188 **NeuroQuantology2022;20(20): 1796-1811**

INTRODUCTION

Overview

Job classification is the process the agency utilizes to determine the appropriate classification level of a role. It allows us to establish internal relativities between specific roles and to set the classification level of a particular set of duties. It is important to note that job classification is the assessment of the role, not the incumbent. I need to apply the classification algorithms of to predict candidate for the job role. Here I am using Naïve bayees and SVM prediction models for classifications. Based on performance evaluation results, I applied classification model at user side for predicting the job recommendations.

Problem Statement

Recruitment Services undertakes all classification assessments within the agency and maintains an unbiased and fair approach to job classification within the agency. Since the summaries provided by the users in their LinkedIn profiles are in the form of a paragraph, it becomes difficult to train the data which is the form of a paragraph. Which indirectly leads to incorrect results and miss classification. This application is developed using Python programming language, for this used python 3.6.7 version software in my Windows 8 operating system. Also used the Django python framework for implementing the web-based application. To store the application-generated data, I used PostgreSQL 9.4 database server.

LITERATURE SURVEY

In data analysis for Human Resources - want to predict someone's job category based on his job summary. Job summaries are created by users to describe their skills and tasks. Our goal is to use NLP to extract information from these free-form text fields and predict the occupation of the user. Summaries provide an interesting platform for testing text understanding and classification techniques. K-NN of word vectors of the title, which proves the power of well-trained word vectors. Adding the tf-idf as a weight could help us improve the classification process. [7]

Which represents the target user's basic characteristics, contains some basic feature descriptions of job positions or candidates and the extracted information from the resumes or homepages. Feedback relevance is the user behavior or actions recorded in the system, including explicit feedback relevance and implicit feedback relevance, such as the numeric rating for items, a binary like/dislike button or textual comments. More information is integrated into user profile,[8] more preferences and interests of target users can be obtained and better matching between jobs and people can be achieved. The recommendatory result is a list of job positions or candidates sorted by the similarity index. The first issue is about job and people matching and user profiling. It's the fundamental procedure in the job recommender system. Recommendation[8] technology mentioned.

The design of a machine learning-based semi-supervised job title classification system for the online job recruitment domain currently in production at CareerBuilder.com and propose enhancements to it. The system leverages a varied collection of classification as well clustering algorithms.[9] Which is composed of both clustering and classification components in a proximity-based classifier setup and results of which are already used across numerous products at CareerBuilder. Then elucidate our long-term goals for job title classification and propose enhancements to the existing system in the form of a two-stage coarse and fine level classifier augmentation to construct a cascade of hierarchical vertical classifiers.Preliminary results are presented using experimental evaluation on real world industrial data.

On basis of this we should build a new recommendation approach and test with real data for employee and staffing data from large companies. Binary representation only: Less attributes used, No perfect measures. Key words search method: One way recommendation. [10] Knowledge acquisition and knowledge engineering problems. No relational aspects are included. Knowledge acquisition and Knowledge engineering problems. Tools and technologies skills excluded. One way recommendation, Hybrid job recommender systems, Content-based job recommender systems. In order to improve the e-recruiting functionality, many recommender systems approaches have been proposed. This article will present a survey of e-recruiting process and existing recommendation approaches for building personalized recommender systems for candidates/job matching. [10].

The present Browse maps, the item-based collaborative filtering platform at LinkedIn. A hybrid of offline/online system, the system computes a latent co-occurrence graph in batch and serves results to users with low-latency. The system's usability and its quick on boarding procedure[11] have enabled many behavior-based recommendation products at LinkedIn in the past few years. The various datasets Browse maps produces are also used in many hybrid recommender systems that combine collaborative filtering and content-based methods. In addition to case studies on how LinkedIn[11] uses the Browse map platform, presented lessons learned in the field over the several years this system has been in production.

Collaborative Filtering

Content-based methods, features of items are abstract and compared with a profile of the user's preference. [12]In other words, this algorithm tries to recommend items that are similar to those that a user liked in the past. It is widely applied in information retrieval (IR). However it performs badly in multimedia field such as music or movie recommendation because it is hard to extract items attributes and obtain user's preference sometimes. CF is a popular recommendation algorithm that bases its predictions and recommendations on the ratings or behaviour of other users in the system. The test results indicate the improved recommender with a rescore is better than a traditional item-based one. Our student job hunting recommender achieved higher precision, recall and F1 score. Furthermore, the recommended jobs are more relevant with students' preferences. [12] To further optimize the recommendation system and ameliorate the sparsity of user profile, some methods of filling users' preference matrix can be utilized.

Student profiling:A student profile is composed of two main attributes, i.e., basic attributes and achievement attributes like individual background and also his educational background, such as home town, gender, university, department, major and courses taken.Student similarity calculation: In this research, we employ a similarity degree representation mechanism by calculating the similarity of attributes between different students with differentweights. Job recommendation: on the basis of recommendation is simple to implement, inspired by the classification algorithm KNN [13], we argue that the most similar graduates will have more influence on the student's job seeking process. Time factor aware job recommendation: This idea is based on an important feature called re ranking, which has been widely adopted in recommender systems and has proven success in improving overall recommendation performance.It is particularly difficult for university students since they normally do not have any work experience and also are unfamiliar with the job market. To deal with the information overload for students during their transition into work, a job recommendation system can be very valuable. As a future work, we intend to conduct a more comprehensive study using students from more than one university to increase the diversity of student background. We also intend to study employer satisfaction while maintaining student expectations since job recommendation is reciprocal, which deserves deep investigation in the future work.

Job Recommender Systems

While the positions of known profiles are assumed to be correct, it should be noted that there are usually multiple advisable positions corresponding to a set of skills. A recommendation system should return a set of most likely positions and all of them can be equally valid. Because of Clustering [14] of Job Positions e job seeking process less effective, more manually conducted and time consuming. User profiling is one of the major issues of these approaches, because retrieving, selecting and handling such data is hard. Both skills and positions are specified by users as free text: many of them are present in multiple in stances across profiles, but the majority of skills are only present in few or single profiles, due e.g. to type or uncommon names. Another issue is the use of different languages across the dataset: many users filled in their profile in Italian due to being their native language, whereas many others used English to target a wider audience. Due to these aspects, the same actual skill or position can be found multiple times with different names. For further research would be to devise a method [14] which fits even better to a recruitment system, for example by testing other machine learning methods such as nearest neighbour classifiers or even exploiting the generated hierarchy. Also the vector representations of profiles, skills and positions could possibly be improved, for example by borrowing suitable weighting schemes from text categorization (Domeniconietal, 2015).

Recommendation Algorithms

Content-based filtering (CBF): In Content-based methods, features of items are abstract and compared with a profile of the user's preference. In other words, this algorithm tries to recommend items that are similar to those that a user liked [15] in the past. It is widely applied in information retrieval(IR). However it performs badly in multimedia field such as music or movie recommendation because it is hard to extract items attributes and obtains user's preference sometimes.

Collaborative Filtering (CF): CF is a popular recommendation algorithm that bases its predictions and recommendations on the ratings or behaviour of other users in the system. There are some drawbacks in Collaborative Filtering as a whole aspect. Collaborative Filtering approaches often suffer from three problems: cold start, scalability and sparsely. These drawbacks [15] are very problematic at times and crucial opportunities can be missed because of this. A lot of improvement and hybrid algorithms need to be implemented alongside CF algorithm. To further optimize the recommendation system, and integrate the system for better performance we keep in check the sparsity of user profile and use some methods of filling user's preference matrix can be utilized implemented alongside CF algorithm.

The features extracted from the candidate's resume as input and finds their categories, further based on the required job description the categorised resume mapped and recommend the most suitable candidate's profile to HR. On pre-processed dataset, we have extracted the features using the Tf-Idf. The cleansed data was imported and feature extraction was carried out using Tf-Idf. The machine learning based classification model or learning algorithms need a fixed size numerical vector as input to process it. ML based classifiers [16] did not process the raw text having variable size in length. Therefore, the texts are converted to a required equal length of vector form during the pre-processing steps. There are few limitations to the model design as of now, but these can be overcome by having more data to train the model. The current limitation of the model are takes CVs in CSV format, but in the real world, the CVs are either in .doc, .pdf, etc. format. Due to the limitation of the data set, the model could not be enhanced to take .doc or .pdf as input, but using a library "extract" this can be achieved. The library can read varied file format and convert them into a single format which can be used as input to the model.

This approach comprises several other smaller recommendations that contribute to problems of

a) Generating serendipitous recommendations
b) Solving the cold-start problem for new jobs and new candidates.

In our approach, we are using Bidirectional Long Short Term Memory with Attention model to capture variation in the progression of candidate job selection. We also generate the blended smaller recommendations that solves cold-start problem and makes the final recommendation [17] serendipitous. A new candidate or a new job may not benefit from machine learning analysis due to the absence of interaction data between the job and the candidate.

Hybridization techniques:

Weighted: Combination of weights from different recommendation components. Switching: Intelligent switching of recommendation [18] systems depending upon the type of request.

Mixed: Recommendations from different recommenders are presented together. Feature Combination: Different pointers from various recommendation engines are selected and parsed into one system. Feature Augmentation: The output of one recommendation system using "Feature Combination" is used as the input for another system which then computes the remaining features.

The Recommendation System needs various features to be able to recommend [18] a job or a candidate. The features for the User are: Education (Highest Degree), Industry, Skills, Location and Current Position. The features for the Jobs are Industry, Skills, Experience, Location, Position, and Salary. Implementing Partner Program of LinkedIn, which requires partnering with LinkedIn for future projects. Using this service developers can get access to every API and data present for any specific user. This will help make skill and endorsements mapping easier and verified.

Competitiveness Analysis - This study divided information technology into five areas as follows: smart content, AI, smart networking, information science and data analysis, and web services or E-commerce. [19] Personality trait analysis - This study conducted a personality analysis using a DISC plane, which was divided into quadrants namely dominance (D), influence (I), steadiness (S), and compliance (C).

Job recommendation analysis - The job recruiter end featured a talent recommendation analysis module. The module also matched the electronic resumes job applicants submitted with the job vacancies provided by job recruiters and recommended the top 20 candidates to job recruiters after compiling the applicants' resumes. [19] using the same algorithm adopted in the job applicant end. DISC Model - Before engaging in the calculation of DISC personality traits extracted from the job applicants' resumes, the researchers pre-processed the data, which consisted in breaking the Chinese words in the documents and filtering meaningless words.

Competitiveness Score - The job applicants' competitiveness score was subsequently obtained. First, the researchers wrote a Web crawler to collect research anddevelopment job vacancies of information technology falling in the five categories classified in this study from online Taiwanese HR agency platforms. Issues with accuracy and reliability. Although AI has come a long way, it is still far from being considered perfect. Too much dependency on certain keywords. AI depends very much on certain keywords to scan through their pile of candidates. Lacks nuance of human judgement. In future studies, the accuracy of the recommendation model should be enhanced, and keywords that commonly appear in job vacancy descriptions should be scored and adopted as reference for weighting, so that the system achieves the performance expected by companies. [19] To enhance the accuracy of the system's recommendation results, researchers can explore in depth the acceptance rate of job applicants after their interview. That is, the actual number of job vacancies that were originally on the recommendation list and later filled by the applicants should be evaluated.

The system tries to make the recruitment process simpler and efficient by integrating text mining and natural language processing techniques. Consists of unique and essential features like study materials, de-duplication process, resume analysis and weight age analysis. The employer can upload study materials while posting the job requirement [20] so that the job seeker will have a fair knowledge of the exact job role. The recommendation of exact profile based on the skill required is processed using collaborative filtering algorithm. The text mining integration of machine learning algorithms, natural language processing, advanced data mining techniques are useful for extracting useful information from large set of data. They involve pre-processing, text mining, sentimental analysis and machine learning techniques to process the input text, usage of association rules and visualization tool to show the output. Productive and good candidates are eliminated in the interview process by the interviewer due to their wrong decisions, wrong prediction of candidate subjective views, personal emotion. Thus because of these parameters right candidate may be missed to get the right opportunity. Application of text mining technique to the candidate resume and projecting the list of candidate resume along with weight age of skills would help the human resource department in identifying the right candidate at short period of time. Thus NLP based resume analysis, interview assessment process of the existing approach cannot provide promising results [20] as candidates invoking top candidates cannot perform or deliver their best during the interview process as it may depend on their physical and mental state during the interview process. The main goal of this portal is to attempt to produce the right graduates based on the industry needs using natural language processing techniques. In future work, we can integrate video based resume format to validate candidate communication and technical skills, which saves time and cost for the recruiter. Also in future, developing job portal as mobile application would have greater scope.

Automated Recommendation is while conducting a search on the web, users are supported by automated recommendation to find and choose the right items that fit their needs, according to people they trust or sharing similar tastes. Automated recommendation is divided into content-based [21] filtering and collaborative filtering. Content-based methods suggest items similar to those a user has selected in the past. Collaborative filtering recommend objects based on the preferences of other users with tastes similar to those of the current user. Text clustering methods Text clustering consist on an unsupervised learning approach that aims to group a given text document set into clusters our groups in a way that documents in a same cluster are more similar between each other. Many techniques are used to accomplish textual content clustering of documents. This model is based on cluster analysis approach, which is a self-organized learning that helps to identify groups of job offers according to the degree of similarity, or dissimilarity between their features.

Our future Work will focus on training and evaluating our model using Word2vec method and k-means clustering algorithms used to capture and represent the context of job profiles. Subsequently, it will be easy to match set of job offers to a given job seeker based on its past interactions toward specific job offers. The dataset that will be used is built from scraping job search websites. [21]

IMPLEMENTATION

Architecture

I discussed the model or methodology of the 'Job Prediction' system. For describing the model, I used the System Architecture diagram mentioned in Figure 1.1. In this architecture, mentioned that the flow of the execution of the system can handle by the two modules, namely, admin and user. The admin and user flow are represented in the architecture separately, like admin flow represented in brown color, and user flow represent in the blue color format in the architecture. Let discuss the individual representation of each feature of the architecture.

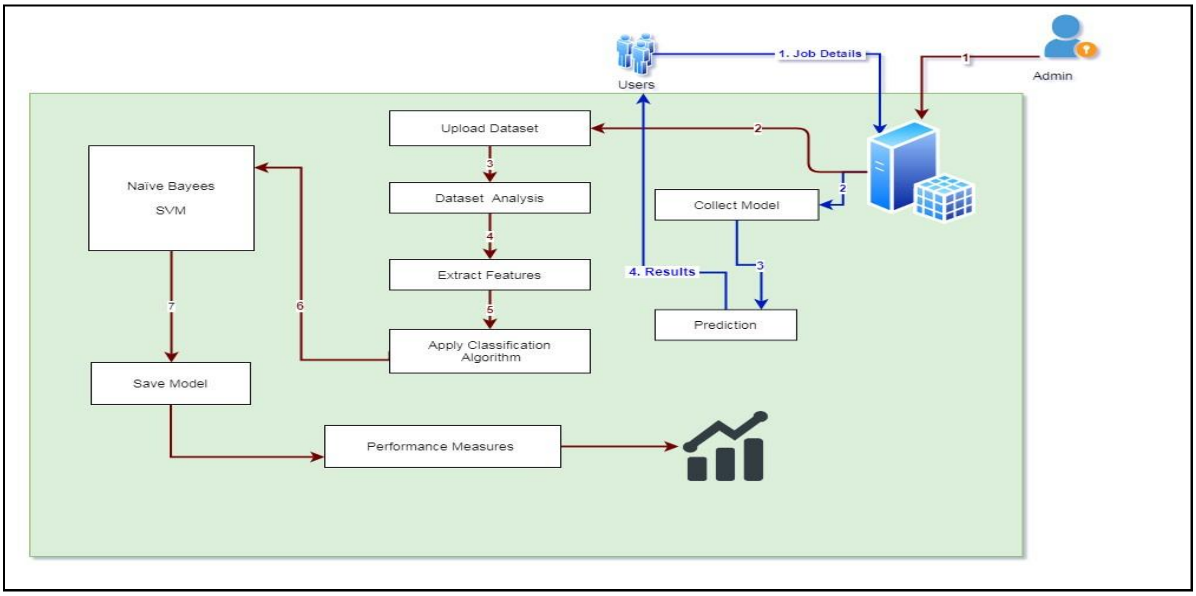

Fig.1.1 System Architecture

The admin is the main user who performs the data analysis part. The admin performs majorly data analysis, data classification, and compares the performances of the classification models. In this project, I have taken two classification models which are mentioned in the following, and used the best accurate algorithm for job prediction on the user side.

- Naive Bayes
- Support Vector Machine

For classification, comparing the performance scores, and prediction of jobs, I have taken the LinkedIn Kaggle dataset [2]. In this section, I mentioned a few observations of the dataset developed using python pandas API.

Dataset Summary

Here, I mentioned list of columns of the dataset.

```
import pandas as pd

r_pd=pd.read_csv('Training.csv')
rs=r_pd.columns

print(list(rs))
```

['Id', 'Title', 'FullDescription', 'LocationRaw', 'LocationNormalized', 'ContractType', 'ContractTime', 'Company', 'Category', 'SourceName']

Total Description of Dataset

Here, discuss about total description like size of the dataset, number of entries of the dataset, length, etc.

```
import pandas as pd

r_pd=pd.read_csv('Training.csv')
rs=r_pd.info()

print(rs)
```

Range Index: 1632 entries, 0 to 1631 Data columns (total 10 columns):

Id	1632 non-null int64

Title	1632 non-null object
Full Description	1632 non-null object
Location Raw	1632 non-null object
Location Normalized	1632 non-null object
Contract Type	609 non-null object
Contract Time	818 non-null object
Company	1030 non-null object
Category	1632 non-null object
Source Name	1632 non-null object

dtypes: int64(1), object(9) memory usage: 127.6+ KB

Sample Data

Here, I represented a one row of sample data of the dataset.

```
import pandas as pd

r.pd=pd.read_csv('Training.csv')
rs=r_pd.head(1).transpose()

print(rs)

```

Id	13656201
Title	Lead Technical Architect, C Banking
Full Description	Lead Technical Architect required for a Tier *...
Location Raw	London

Location Normalized	London
Contract Type	Na N
Contract Time	permanent
Company	Scope AT Limited

Category	IT Jobs
Source Name	jobserve.com

Dataset Analysis

Here, I cauterize the data with job categories and designed a PIE graph for representing the category size of the dataset. In figure 1.2, I mentioned the PIE graph for representing the categories size of the dataset.

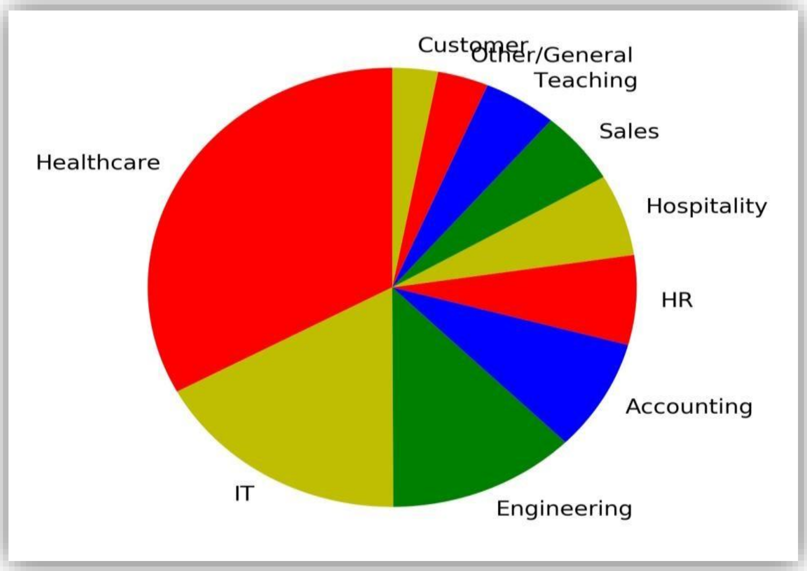

Fig.1.2 Dataset Analysis

Classification Models

In this project, I used two Machine Learning algorithms for classifying LinkedIn job profile dataset for prediction jobs. Main project research questions are 'Which classifier is more accurate to predict job request ?', and 'How does ML algorithm can classify the test data?'. Here, I used two Machine Learning algorithms of supervised, which are mentioned in the following.

- Naïve Bayes algorithm
- Support Vector Machine algorithm

1. Naïve Bayes algorithm

Naïve Bayes algorithm is a popular supervised algorithm that is an enhanced version of the Bayes theorem [3]. This algorithm is most popular in Spam Filter applications, Encyclopedia, etc. This algorithm works like the availability of the input in class is unrelated to the availability in another class. It takes high computation when dataset volume is high and gets more accuracy. Based on the Naïve Bayes theorem, many advanced techniques were proposed like the Bernoulli NB algorithm, Multinomial NB algorithm, etc. In my proposed system, I used the Bernoulli Naïve Bayes algorithm. In the following coding snippet, I mentioned Bernoulli Naïve Bayes algorithm implementation of the training dataset.

2. Support Vector Machine algorithm

Support Vector Machine (SVM) algorithm is also a popular Machine Learning algorithm in the supervised category. This is a very expensive algorithm in terms of complexity and computation.

Based on the labeled data, this algorithm can identify the classes of the impute based on its own agenda. The main agenda of SVM is to separate the classes with the hyper line concept. After separation of the classes with hyper lane, it can again identify another hyper line for better accurate separation of the data. It uses the Kernal function to calculate the hyper line.

TFIdfVectorizer

All the Machine Learning algorithms takes only numerical inputs for mathematical approaches.

These algorithms can't understand the text data (String input). I need to convert the text data into numerical. For converting text data to numerical data have two approaches in preprocessing concepts, namely, CountVectorizer and TfidfVectorizer. The CountVectorizer concept is depends on the count of the words in the document, but in TfidfVectorizer calculate the TF-IDF score for each word of the document. Let see procedure and an example of TfidfVectorizer. In this project, for Naïve Bayes and SVM algorithms I used TfidfVectorizer for classification of dataset.

$$\text{TF}= \frac{\text{Count of a word } w \text{ in a document}}{\text{Total No.of words in a document}}$$

$$\text{IDF} = \frac{\text{Total Documents Count}}{\text{Document Frequency of word present}}$$

$$\text{TF-IDF}=\text{TF} * \text{IDF}$$

For example,
Documents are 'President of Britain, 'Britain queen', 'India president

TfidfVectorizer

Doc	*Britain*	*India*	*Of*	*President*	*queen*
0	*0.517*	*0.000*	*0.680*	*0.517*	*0.000*
1	*0.605*	*0.000*	*0.000*	*0.000*	*0.795*
2	*0.000*	*0.795*	*0.000*	*0.605*	*0.000*

Build Models

After classifying the data, with the classification object I can predict the input data. But user side, the system is required to predict the data n number of times. For predicting the data multiple times with a classification object, this object should be available continuously. Based on these requirements, I saved the training classification object into a physical file, which can available. Using 'Pickle' API, I saved classification objects into .sav files. In Fig 1.3, I represented the model files of NB and SVM algorithms.

Performance Measure

For calculating the performance of the algorithms I have taken Accuracy Score to compare performance measures. For calculating accuracy, I have taken 30% of dataset for testing. To calculate the Accuracy score, I need to identify the values of the four parameters. Those are True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). In the following, I mentioned accuracy equation.

Accuracy:

$$\text{Accuracy} = \frac{(TP) + (TN)}{\text{Total inputs } (TP + TN + FP + FN)}$$

After identification of the accuracy scores of algorithms, I represented in graph.

EXPERIMENT RESULTS

Dataset Analysis

Admin can see the description of dataset in a pie graph.

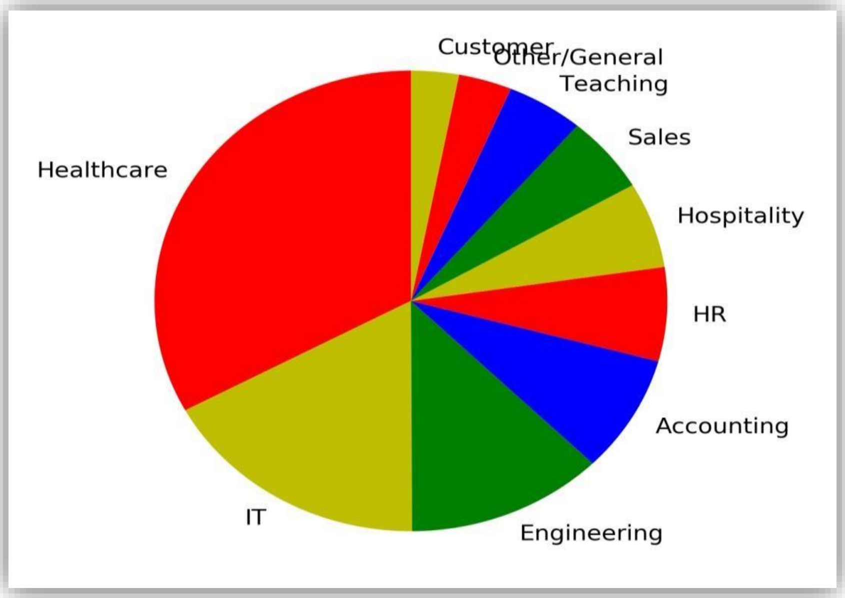

Fig 1.3: Dataset Analysis

View Accuracies

The admin can see the accuracies of the algorithms in a tabular format.


View Accuracies

Algorithm Name	Accuracy
Naive Bayees	0.900302114803625
SVM	0.848942598187311

View Accuracy Graph

Fig 1.4: View Accuracies

View Accuracy Graph

The admin can see the accuracy graph of the algorithms in a bar chart format.

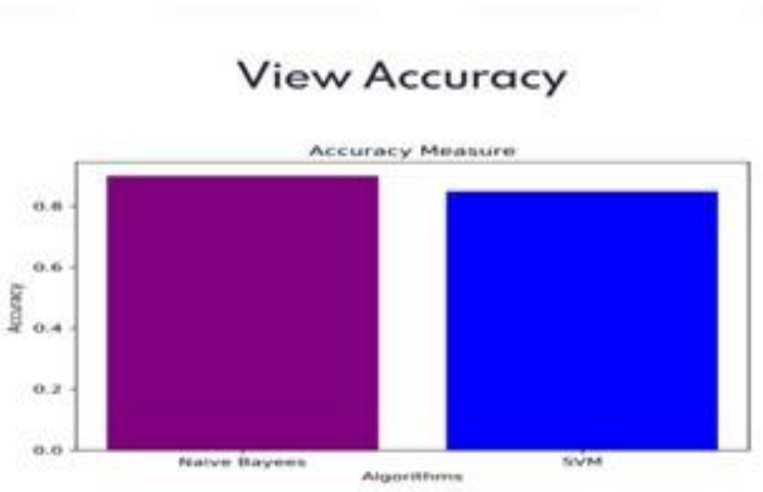

Fig 1.5: View Accuracy graph

Job Results

RESULTS

<table>
<tr><th>Title</th><th>Description</th><th>Location</th><th>Company</th></tr>
<tr><td>Developer x**** (C or Java or C++)</td><td>Developer x **** C or C++ or JAVA Central Southampton ****k****k bensKey skills: Academic or</td><td>Southampton</td><td></td></tr>
<tr><td>Senior Software Engineer Java or NET</td><td>Senior Software Engineer Java or .NET (C, ASP.NET) Belfast **** **** Benefits A leading global organisation</td><td>Belfast</td><td>ARRAY</td></tr>
<tr><td>Technical Project Manager Agile and Java J****EE</td><td>Key project management / software development role where you will be responsible for the overall</td><td>Surrey</td><td>JOBG8</td></tr>
<tr><td>QA Automation Engineer | Software Test engineer Java, Web</td><td>QA Automation Engineer | Software Test engineer Java, Web Salary: ****K Bonus (10%) Excellent Benefits</td><td>London</td><td>JOBG8</td></tr>
<tr><td>Front End Developer XHTML, CSS, JavaScript, Lamp, PHP, WordPress</td><td>This is a fantastic opportunity for a Front End Developer to join a leading Marketing Agency in</td><td>London</td><td>JOBG8</td></tr>
</table>

Fig 1.6: Job Prediction Results

The user can see the job results by Naïve Bayes algorithm.

CONCLUSION AND FUTURE WORK

Job recommendation is a fundamentally different problem from traditional recommender systems for books, products, or movies. For this we need to consider job value factors such as technologies, skills, qualifications, etc. are taken into consideration. Our goal is to use extract information from these free-form text fields and predict the occupation of the user. Classification models are deals with only numerical data, but text fields like job technologies can't handle by the ML algorithms. For this used TFIDFVectorizer with ML algorithms of Naïve Bayes and Support Vector Machine (SVM) algorithm for building models. In these results I achieved almost 90% accuracy for Naïve Bayes algorithm, In the future work, prefer to perform job classification using Deep Learning models like CNN, R-CNN, etc.

REFERENCES

[1] Mirko Polato and Fabio Aiolli. 2016. A preliminary study on a recommender system for the job recommendation challenge. In Proceedings of the Recommender Systems Challenge (RecSys Challenge '16). ACM.

[2] Linkedin dataset, Available at: https://www.kaggle.com/chadalee/text-analytics-explainedjob-description-data/data.

[3] Analytics Vidhya, Available at: https://www.analyticsvidhya.com/blog/2017/09/naivebayes-explained/

[4] Analytics Vidhya, Available at: https://www.analyticsvidhya.com/blog/2017/ 09/understaing-support-vector-machine-example-code/

[5] Jose Ignacio Honrado, Oscar Huarte, Cesar Jimenez, Sebastian Ortega, Jose R. PerezAguera, Joaquin Perez-Iglesias, Alvaro Polo, and Gabriel Rodriguez. 2016. Jobandtalent at RecSys Challenge 2016. In Proceedings of the Recommender Systems Challenge (RecSys Challenge '16). ACM.

[6] Kuan Liu, Xing Shi, Anoop Kumar, Linhong Zhu, and Prem Natarajan. 2016. Temporal learning and sequence modeling for a job recommender system. In Proceedings of the Recommender Systems Challenge (RecSys Challenge '16). ACM.

[7] Eric Boucher, Clement Renault Stanford University,Ykhlef2."Job Classification Based on LinkedIn Summaries" CS 224D,2012.

[8] Zheng Siting, Hong Wenxing*, Zhang Ning, Yang Fan School of Information Science and Technology Xiamen University, "Job Recommender Systems: A Survey", Accepted 14 July, 2012.

[9] FaizanJaved, Matt McNair, Ferosh Jacob, Meng ZhaoClassification R&D, CareerBuilder.com, 5550 Peachtree Parkway, Greater Atlanta Area, GA 30092, USA."Towards a Job Title Classification System."

[10] Shaha T. Al-Otaibi1* and Mourad Ykhlef "A survey of job recommender systems", College of Computer and Information Sciences, Princess Nora BintAbdulrahman University, Accepted 3 August, 2012.

[11] Lili Wu,Sam Shah, Sean Choi, Mitul Tiwari ,Christian Posse. The Browsemaps: "Collaborative Filtering at LinkedIn."

[12] Yingya Zhang, Cheng Yang, Zhixiang Niu Information Engineering School Communication University, "A Research of Job Recommendation System Based on Collaborative Filtering", Accepted by 2014.

[13] Rui LIU, Wenge RONG, Yuanxin OUYANG, Zhang XIONG, "A hierarchical similarity based job recommendation service framework for university students", Accepted April 13, 2016.

[14] Giacomo Domeniconi, Gianluca Moro, Andrea Pagliarani, Karin Pasini and Roberto Pasolini, "Job Recommendation from Semantic Similarity of LinkedIn User's Skills", Accepted by 2016.

[15] Vinay Desai, Dheeraj Bahl, Shreekumar Vibhandik, Isra Fatma, "Implementation of an Automated Job Recommendation System Based on Candidate Profiles", Accepted 05 May, 2017.

[16] Pradeep Kumar Roya, Sarabjeet Singh Chowdharyb, Rocky Bhatiab, "A Machine Learning approach for automation of Resume Recommendation system", Accepted by 2019.

[17] Amber Nigam1,2*,Aakash Roy1*, Hartaran Singh1†, Harsimran Waila, “Job Recommendation through Progression of Job Selection”, Accepted by 2019.

[18] Harsh Jain, Misha Kakkar, “Job Recommendation System based on Machine Learning and Data Mining Techniques using RESTful API and Android IDE”, Accepted by 2019.

[19] Yi-Chi Chou, Han-Yen Yu, Digital Education Institute Institute for Information Industry, “Based on the application of AI technology in resume analysis and job recommendation”, Accepted by 2020.

[20] Jaichandran, Leelavathy, Kanaga Suba Raja, Pranav Kumar, Sailendra Kumar Majhi and Vikas Singh Thakur, “Job Portal Resume Evaluaton System Using Text Mining And Natural Language Processing”, Issue 4, 2020.

[21] D. Mhamdi, R. Moulouki, M. Y. El Ghoumari, M. Azzouazi, L. Moussaid, “Job Recommendation based on Job Profile Clustering and Job Seeker Behavior”, August 9-12, 2020.