INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 6.078
(Volume 6, Issue 4)
Available online at: https://www.ijariit.com

Candidate talent assessment through recommender systems using machine learning techniques

Swathi V.
swathivenkatesh.98@gmail.com
BNM Institute of Technology, Bengaluru, Karnataka

Suraksha S. Tasgaonkar
surakshatasgaonkar@outlook.com
BNM Institute of Technology, Bengaluru, Karnataka

Shubhashri T. V.
shubhz1998@gmail.com
BNM Institute of Technology, Bengaluru, Karnataka

Prarthana T. V.
prarthanatv@bnmit.in
BNM Institute of Technology, Bengaluru, Karnataka

ABSTRACT

The world is moving towards complete automation where most of the systems are being automated, one of the examples of this trend is the automation of IT industry. The candidate recommender system helps in the selection of candidates for a company. The candidate recommender system looks into the various profile of the candidates chooses a Candidate whose profiles best matches that of the company and the job.

Keywords: *Recommender Systems, Machine Learning, Candidate Recommender, K-Nn, Matrix Factorization, Collaborative Filtering*

1. LITERATURE SURVEY

Talent Assessment and Acquisition portal recommends candidates whose profile matches that of the organization. One can also view the profiles of various candidates and also set up the job interviews for the candidate. If the company chooses to select the candidate for the further process, then it digs deeper into the candidates profile and recommends him for the closest match available in terms of job opening in the company. Upon qualifying for the job, the details such as the job role and CTC can also be assigned. Furthermore, the organization can update an opening for a particular job. The employee recommendation system using various recommendation techniques like simple matching and then collaborative and content based filtering. Similarity metrics are used to calculate how much similar all the items are to each other in the matrix. Comparison is made between different recommendation approaches. The objective of the project is to develop a system that helps the HR of the company to select candidates without ruse. To ease the process of interview for the recruiters and reduce their burden. The system learns from the past preferences of the recruiter and then recommends candidates.

Anitha Anandhan, Liyana Shubhi et al (2018) identified the suitable item or information in the Internet becomes challenging due to information overload. In general, the four different approaches used in developing the Recommender Systems include Content-Based (CB) filtering; Collaborative Filtering (CF), Hybrid- Based (HB) filtering, and Knowledge- Based (KB) filtering. Various techniques are used in social media RS to explore and analyse large quantities of data in the form of patterns and rules. Bayesian Network, and logistic regression techniques were used for CB filtering. Among the data mining techniques the most widely used for CF are clustering, kNN, matrix factorization, link analysis, decision tree, and association rule. Clustering, matrix factorization, kNN techniques, fuzzy techniques are mostly used in HB filtering.

Zwiwei Guo, Wenru Zeng et al (2019) identified ties among people have been much more closer, making recommendations for groups of users became a more general demand, which facilitates the prevalence of Group Recommender System (GRS). This paper proposes GRS-PR, an enhanced group recommender system by exploiting preference relation. First, a preference relation-based multivariate extreme learning machine model is formulated to predict unknown preference relations in candidate items. Second, on the basis of predicted results, borda-voting rule is employed to generate recommendation results from candidate items. In addition, efficiency, parameter sensitivity, and sparsity tolerance of the GRS- PR are evaluated through a set of experiments. In view of above analysis, this paper proposes an enhanced Group Recommender System by exploiting Preference Relation (GRS-PR). First, definition of preference relation is announced. Then, based upon preference relations of item pairs in training set, learning machine model is formulated to predict unknown preference relations in candidate items.

Shiza Hammad, Iqra Abbasi proposes recommendation system considers interviewers preferences and the sequence in which they wanted their preferred candidates to appear in the final selection list. The proposed recommendation system requires only interviewers preference vectors (v) in addition to other known variables i.e. number of posts (K), number of interviewers (n), total number of candidates (m) and their objective evaluation scores. The proposed recommendation system uses two algorithms, the Hungarian Aggregated Method (HAM) and the Greedy Aggregated Method (GRAM) for computing the final decision vector (r). The proposed system requires four inputs i.e. total number of interviewers (n), total number of candidates (m), total number of positions (k) and interviewer preference vector (v). The input variables n, m and k are non-negative integer values. Whereas, interviewer preference vector (v) is of the form v = <$Item_i$, $Item_{i+1}$,...,$Item_n$> where i = 1 to k and $Item_1$ represents the candidate at the first position in the interviewers preference. The result generated by the proposed system will be a vector (r) similar as vector (v) containing final selected list of candidates.

Hafed Zrzour, Ziad Al-Sharif, introduced a new collaborative filtering recommendation algorithm based on the dimensionality reduction and clustering techniques. The aim is to improve the performance of recommender systems and to overcome the problems of sparsity and cold-start where new users enters into recommender systems and initial ratings is not assigned for those users, it also overcomes scalability issues. The k-means algorithm and Singular Value Decomposition (SVD) are both used to cluster similar users and reduce the dimensionality, respectively. SVD is one of the dimensionality reduction techniques that are recognized for their capacity to improve the scalability of recommender systems the experimental results show that the method significantly improves the performance of the recommendation systems. A key contribution of work is to construct, in two main phases, an effective recommender system that can generate accurate recommendations regardless of the dataset size. The first stage is called offline model creation. In this stage, the model of recommendation is created by clustering the users' ratings following their preferences, reducing the dimensions of data and then calculating the similarities. The k-means algorithm and SVD technique are both used in this stage to cluster similar users and reduce the dimensionality, respectively.

Shaokang Dong ,Zijan Lei, provided details to make every user conveniently have access to his or her most interested jobs and candidates (recommendation items) in the current employment market, the recruitment networks need to meet the demand of fast and accurate recommendation. But a key challenge is that the total items may have a large quantity in the big data scenarios. And another problem is that the personalization of different users is diverse. In order to handle these challenges, this paper proposes a mining and prediction system for job and candidate recommendation with contextual online learning. It predicts a proper item by utilizing the feedback reward of previous users in the nearby context region. Monte-Carlo Tree Search (MCTS) method is used in which the similar items can be amalgamated into a cluster to reduce the computing load. The Monte-Carlo Tree only from the root, which contains all the items. And in the exploration process, expanding the leaf node into two specific nodes to get more accurate analysis as long as it has been selected enough times, which can address the main challenge of large-scale items to make it more applicable in the recruitment networks. Besides that, the cold start problem can also be relieved in this algorithm because the users are clustered depending on the context information and can analyze the feedback results of previous users who are in the nearby context region. The comparison between proposed work and the other state-of-the-art algorithms. The performance of the algorithm is evaluated on the database, which is extracted from the Work4 test data. The Work4 test data contains the candidates' profile information based on Facebook and recruiting jobs based on LinkedIn, which is a business and employment-oriented social networking service that operates via websites.

Wenbo Chen, Shaokang Dong, Shimin Gong, used Personalized Professional Network Recommenders (PNRs) are required to promptly make accurate recommendations to the job seekers and employers and provide real-timely update to system's policy to maximize users' satisfaction. To achieve personalization, user's explicit information (e.g., candidates' age, gender and experiences and jobs' type, salary and key skills) should be effectively used.

However, recommendations only based on explicit information could result in high risks such as a higher number of dissatisfied job seekers and employers. This problem is solved using implicit information. Implicit information consists of all signals about users' interests that can be deduced by their online behaviors such as the pages they visited, the time they stayed on a specific page and the items they saved for revisiting. Contextual information emulates like user preferences, browser records and information obtained during registration of the user is taken into account for recommendation.

2. PROPOSED SYSTEM

The data of candidates i.e, the resume is collected. The dataset for the system is taken in the form of csv files as well as JSON files. Both these files contain candidate resumes. Additionally, candidate resumes are collected from a website, where candidates upload their resumes. The resume is filtered for the appropriate data (skills, past experiences, education and also their projects). This is fed to the recommender system that applies appropriate machine learning algorithms to recommend suitable candidates based on company's profile. The resumes of candidates who are already recommended and selected by the company are used for further recommendations. The website will also help conduct interviews for the recommended candidates and monitor the progress of the candidate through various rounds of interview.

The methodology below provides a brief overview:

a) Consider a dataset of resumes: Obtain the required data
b) Create a labelled data with the dataset obtained.
c) Apply the machine learning algorithms for recommendation.
d) Obtain the recommended candidates and display on the frontend of the browser.
e) Allow the user to select the final list of selected candidates; this will be stored in the database.
f) Use this data of candidates selected stored in the database for further recommendations.

The data is taken from Kaggle where the dataset contains resumes of various candidates. The dataset consists of the id, type of job post and resume.

Data is also taken from a website that is created solely for the purpose of recommendation as well as selecting candidates where the candidates can create their resume if they do not have one.

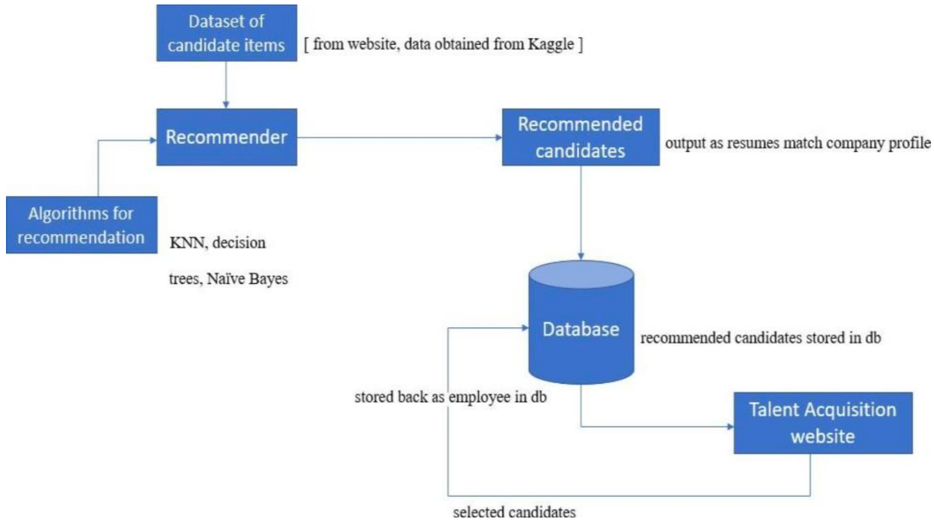


Fig. 1: Proposed System of Candidate Recommender System

3. FEATURE SELECTION

When recommendation for a particular job role it is required that we extract features such as skills, previous experiences, projects, certifications and details regarding a candidates study. Skills for our simplicity is classified into beginner, intermediate and expert, which have their own weights in terms of numbers. These are considered when forming the final dataset. Projects and certifications are also taken into consideration when forming the final dataset. The years of experience in a particular company and a particular field also add weightage to the dataset.

4. SYSTEM DESIGN

The algorithms used to predict whether a candidate is suitable for are a job role are: *K-NN* and matrix factorization. K-NN works by finding the distances between a query and all the examples in the data, selecting the specified number examples (K) closest to the query, then votes for the most frequent label (in the case of classification) or averages the labels (in the case of regression). Its purpose is to use a database in which the data points are separated into several classes to predict the classification of a new sample point. Just for reference, this is "where" K-NN is positioned in the algorithm list of scikit learn. K-NN is a lazy learner because it doesn't learn a discriminative function from the training data but "memorizes" the training dataset instead. For example, the logistic regression algorithm learns its model weights (parameters) during training time.

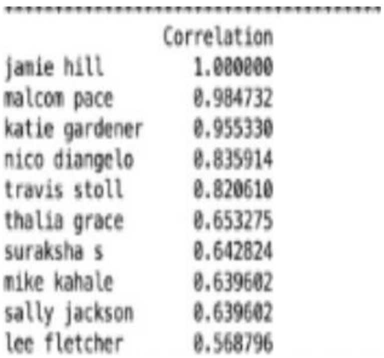
1) The steps to apply K- NN algorithm are: 1.Determine parameter K = number of nearest neighbors.
2) Calculate the distance between the query-instance and all the training samples.
3) Sort the distance and determine nearest neighbors based on the K-th minimum distance.

```
   Developer  Java  Machine Learning  Python  Decision
0          1     2                 0       4  unselected
1          0     1                 0       4  unselected
2          1     0                 0       0  unselected
3          1     0                 0       0  unselected
4          0     0                 1       0  unselected
5          1     2                 0       4  unselected
 Train Set: 10
 Test Set: 6

The predictions are:
  Name:  Robert, predicted: 'selected'
  Name:  Milan, predicted: 'selected'
  Name:  George, predicted: 'unselected'
  Name:  Lynn, predicted: 'unselected'
  Name:  Harry, predicted: 'unselected'
  Name:  Ron Hardy, predicted: 'selected'
```

Fig. 2: Output of *K-NN* algorithm

When we consider Matrix factorization the candidate dataset matrix each row represents each users, while each column represents different skills. Obviously the matrix will be sparse since not everyone has all skills (we all have different skills).One strength of matrix factorization is the fact that it can incorporate implicit feedback, information that are not directly given but can be derived by analyzing user behavior. Using this strength we can estimate if a candidate is suitable for a particular job. And if that estimated rating is high, we can recommend that candidate to the job post. We find the candidate who has highest suitability for the job post and try to correlate other candidate to this candidate.



	Correlation
jamie hill	1.000000
malcom pace	0.984732
katie gardener	0.955330
nico diangelo	0.835914
travis stoll	0.820610
thalia grace	0.653275
suraksha s	0.642824
mike kahale	0.639602
sally jackson	0.639602
lee fletcher	0.568796

Fig. 3: Output of Matrix factorization

5. RESULT ANALYSIS

The time that is used by Matrix factorization and the KNN algorithm run on the same input values. And there is almost the same output obtained by both the algorithms. Thus, the time complexity can be included as one of the parameters for plotting the graph. As seen in the graph K-NN works much faster and is a better algorithm when matrix factorization is considered.

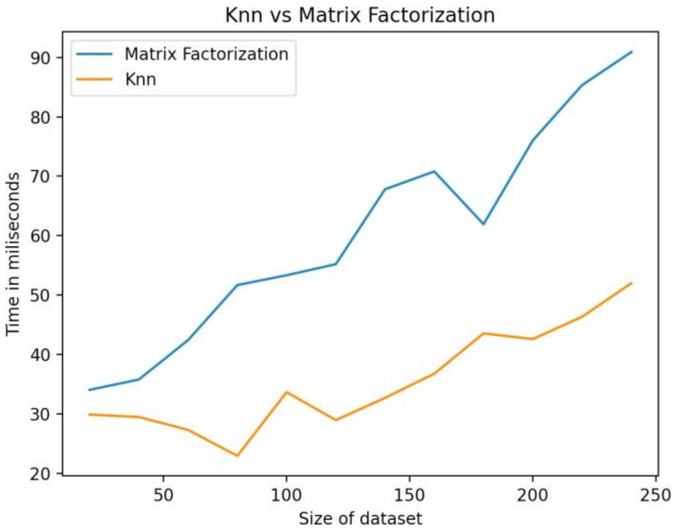


Fig. 4: Analyzing performance of K-NN and matrix factorization algorithms

6. CONCLUSION

Recommender systems are a powerful new technology for extracting additional value for a business from its user databases. These systems help users find items they want to buy from a business. Recommender systems benefit users by enabling them to find items they like. Conversely, they help the business by generating more sales. Recommender systems are rapidly becoming a crucial tool in E- commerce on the Web. Recommender systems are being stressed by the huge volume of user data in existing corporate databases, and will be stressed even more by the increasing volume of user data available on the Web. The Candidate Recommendation system recommends candidates that are suitable for a particular job without having to go through all the candidates on a job-searching platform. E- commerce platforms like Amazon and social networking sites like Facebook inspired the Candidate Recommendation. Talent Assessment and Acquisition portal recommends candidates whose profile matches that of the organization. If the company chooses to select the candidate for the further process, then it digs deeper into the candidates profile and recommends him for the closest match available in terms of the job opening in the company. Upon qualifying for the job, the details such as the job role and CTC can also be assigned.

7. REFERENCES

[1] Zwiwei Guo, Wenru Zeng and Yu Sen,"An Enhanced Group Recommender System by Exploiting Preference Relation."IEEE Access, Feb 5,2019
[2] Shiza Hammad, Iqra Abbasi, "A Recommendation System Using Interviewers Preferences for Ranked Candidate Selection" IEEE – International conference on Frontiers on Information Technology Sep 2019.
[3] Hafed Zrzour, Ziad Al-Sharif, "A New Collaborative Filtering Recommendation Algorithm Based on Dimensionality Reduction and Clustering Techniques", IEEE Access Vol 4 Jan 2019.
[4] Shaokang Dong ,Zijan Lei, "Job and Candidate Recommendation with Big Data Support:A Contextual Online Learning Algorithm" ,IEEE Access, Vol 36 Feb 2019
[5] Wenbo Chen, Shaokang Dong, Shimin Gong "Tree-Based Contextual Learning for Online Job or Candidate Recommendation With Big Data Support in Professional Social Networks", IEEE Date of publication Nov 29, 2019
[6] Anitha Anandhan, Liyana Shubhi, Maizatul Akmar Ismail, Ghulam Mujtaba "Social Media Recommender Systems: Review and Open Research Issues", IEEE Access, Volume 6 Feb 2019.

