Applying Data Mining Techniques in Job Recommender System for Considering Candidate Job Preferences

Anika Gupta, Dr. Deepak Garg

Computer Science and Engineering Department
Thapar University, Patiala, India
anikagupta2010@gmail.com, dgarg@thapar.edu

***Abstract*—Job recommender systems are desired to attain a high level of accuracy while making the predictions which are relevant to the customer, as it becomes a very tedious task to explore thousands of jobs, posted on the web, periodically. Although a lot of job recommender systems exist that uses different strategies , here efforts have been put to make the job recommendations on the basis of candidate's profile matching as well as preserving candidate's job behavior or preferences. Firstly, rules predicting the general preferences of the different user groups are mined. Then the job recommendations to the target candidate are made on the basis of content based matching as well as candidate preferences, which are preserved either in the form of mined rules or obtained by candidate's own applied jobs history. Through this technique a significant level of accuracy has been achieved over other basic methods of job recommendations.**

Keywords*— Data mining, Decision Tree, Classification Rules, Content Based similarity, Job recommendations.

I. INTRODUCTION

Recommender systems are being used in almost every internet based ecommerce websites. However the type of recommendations provided may vary according to the domain of its usage. For example, in an online shopping site for clothing, it will be more favorable to provide generalized recommendations regarding the latest trends and fashion in the market, as most of the users are expected to go along that way. However, this case is little bit different in e-recruitment sites. Here, it will be favorable to provide more personalized and profile based job recommendations. In job recommender systems, there are varieties of customers/ candidates, having different education level, experience and skills. Based on their respective background details, each one expects to get only those job recommendations which are highly relevant for the respective candidate.

Talking about the human nature, as one is in the youth stage, the more enthusiastic he/she is, and may be ready to take any risk. But as he/ she grow older, a certain level of maturity is gained and one is more tilted towards stable and more promising and less risky decisions in his/her life. The same is applicable while taking decisions regarding one's source of living. While exploring the job recommender systems, interesting facts and figures were obtained regarding the nature of these job applicants. These job applicants, who belong to different age groups, gender etc, show a certain level of similarity, in nature, while applying for jobs. This only formed the basis of the research in this field. Here it is tried to explore these generalized job behavior of candidates having different genre, in the form of classification rules. Later, these mined rules were applied, for providing initial recommendations, to the new candidate according to his genre.

Also, two candidates having similar looking profiles may have different job tastes. Here, job taste can be defined as the preference criterion considered before applying for a particular job. For one, preference can be of getting a job in higher company, as opposed to the other who may be interested in having a job which offers higher salary. Considering this, the second phase of recommendations, are provided to the respective customer according to his/ her job taste or preferences. These job preferences are extracted from the already applied jobs basket of the candidate.

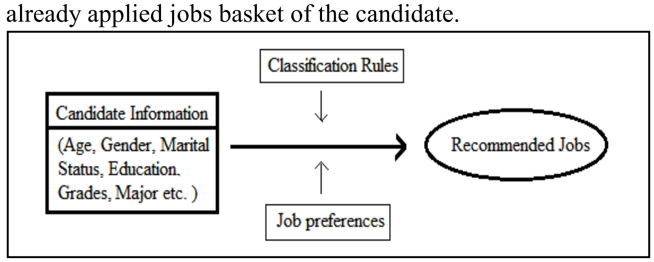

Fig. 1. Proposed Job recommender system framework

II. LITERATURE SURVEY

A lot of research has been carried out in the field of job recommender systems. A variety of job recommender systems already exist that try to explore one or the other aspect of the information by applying different methodologies [1] [2].

CASPER system tries to reduce the information overload of jobs, by providing personalized recommendations to the candidate. CASPER PCF uses a two stage process where in first stage, the similarity between the candidate's query and the jobs, which is in turn done on the server-side. In the second stage, on the client side, retrieved jobs relevance is calculated according to the target user profile and the jobs are finally sorted in the order of their relevance [3].

Proactive is an adaptive system that provides four different interfaces to capture the candidate's job tastes/ preferences. It integrates preferred jobs, recommended jobs, advanced search and most recent jobs in one system for providing efficiency in job recommender systems [4]. The bilateral job recommender system, helps to provide 2-sided recommendations, where first is the CV recommender and next is the Job recommender, by integrating them into one system [5]. Reciprocal recommender systems are also in place where a reciprocal value is calculated considering the candidate's resume information and candidate's interaction history. The final recommendations are provided by ranking based on the reciprocal scores [6].

Hong *et al* tried to cluster the candidates into three major groups and apply different recommendation approaches for each group cluster [7]. Paparrizos *et al* tried to learn from the past transition history of the candidate to provide future job recommendations [8]. Hong *et al* provided a solution to the job recommendations, from two basic perspectives of time and dimensionality. In this they dynamically updated the candidate's profile over the interested and non-interested feature sets, extracted over a time period, through the information gain concept [9]. Chien and Chen mined rules for effective personnel selection. They used the candidate's demographic data for predicting his/her work behavior (Job performance, retention and turnover) [10].

III. Problem Definition

A. Problem Formulation

A job recommender system is expected to provide recommendations in 2 ways: 1) recommending most eligible candidates for the specified job, to the recruiters and 2) recommending jobs to the aspiring candidates according to their matching profiles.

The focus of this paper is the second part only i.e. to recommend jobs to the candidates according to their matching profiles. However there can be seen some gap between the existing systems. Here an example is shown:

Suppose a profile of a candidate can be represented as follows: {age, gender, marital status, education, major, education level, experience, current location, skills possessed}. And job as: {field, required education, required experience, required skills, level of company (A (highest), B, C, D), position level offered by the company (A (top positions), B, C, D), pay-scale (High (H), Medium (M), Low (L)), Job Location}

Example 1: There are 2 candidates with following profiles:

1) {27, male, unmarried, graduate, Computer Science, 65%, 6, New Delhi, (Python, Oracle, Machine Learning, English)}

2) {35, male, married, masters, Computer Science, 65%, 7, New Delhi, (Python, Oracle, Machine Learning, English)}

And there are 2 jobs with following requirements:

1) {CSE, graduate, 5, {Python, Machine Learning, English}, B, C, M, New Delhi}
2) {CSE, graduate, 5, {Python, Machine Learning, English}, B, B, M, Bangalore}.

Now, if both candidates are given option to select their prioritized jobs then the case may be that 1st candidate selects 2nd job as priority job whereas 2nd candidate may give priority to 1st job.

Explanation: Although both the candidates possess almost equivalent profiles, both also qualify for both the jobs, still their preferences regarding the jobs are different. The 1st candidate may have chosen for the 2nd job as he considered higher position (level B) and does not have a location constraint in his personal life. However, 2nd candidate, considering his age and marital status, he is normally expected to go with the first job as, amongst both the jobs, the only difference is of the position offered. Else the package offered, company level rest all is same. And also the location of 1st company is similar to his current location. Hence, he may compromise for the position offered and decide to choose 1st job as his preference job.

So, the efforts have been put to judge the gap between the candidate's choices, belonging to different groups, regarding the selection of different offered jobs. In this paper, it is tried to foresee the customer preferences regarding the jobs on four basic parameters of company preferences, position offered, pay-scale offered and job location. Instead of tracking the past history of the candidate, his current likings/preferences are focused upon. For a new customer, the system tries to impose the general job preferences, obtained through mined rules, according to the age-group, gender etc. to which the candidate belongs and as the candidate becomes active, his/her own job preferences are taken into consideration.

B. Terms Used

The terms that are used in this paper, that need some explanation are:

Grades: Marks scored by the candidate in his last acquired degree.
Major: Specialization field or discipline of the candidate.
Experience: Candidate's experience in years.
Skills: Extra knowledge/ skills possessed by the candidate.
Current employment status: Information regarding candidate's present employer, pay-scale, position etc.
Employer/Company: Company, that is offering the job.
Industry field: major/discipline in which the job is offered.
Position Offered: position offered for the job.
Pay Scale: Salary offered for the job.
Job Location: after getting the job, where the candidate will be joining.
Cosine similarity: A measure of similarity between 2 vectors (Here it refers to job and candidate).
Preference matrices: These matrices represent the candidate's preferences in four different directions: company selection, position selection, pay-scale selection and job location selection.
Rules Weight: This refers to the weights assigned to the jobs after applying the group/candidate's own preferences for recommending jobs to the candidate.

Final Weight: This represents the weighted score of a particular job, obtained after combining weighted sum of cosine similarity and rules weight.

C. *Assumptions*

1. It is assumed that all the text categorization is already in place and we have discrete and well labeled values that are easily understandable by the system.
2. It is assumed that either the candidate or the system itself updates the age, experience, education and skills field. Either the candidate can be alerted periodically for updation of these fields else automatic updation can be done with the help of Dynamic modification and extraction method used in [9]. It uses TF-IDF value for feature extraction and information gain with threshold value for feature addition.

IV. Methodology

A. *Feature Selection*

The features that were found relevant in a job scenario belong to 2 major categories: Candidate and Job. For candidate the features that were considered for judging his/ her behavior are: Age, Gender, Marital Status, Education, Grade, Major, Experience, Skills, Current Location, and Current employment status (if any). And the features relating to the job are: Required Qualification and Experience, Skills requirement, Employer or the Company, Industry field, Position Offered, Pay Scale and Location.

B. *Data Categorization*

As the objective was to find out the criteria on which the candidate, belonging to different age group, marital status, gender, education level, grades etc, focuses for selecting the offered job, the complete categorization or generalization was done. The candidate data as well as company data both are categorized into different groups for finding out the candidate's behavior belonging to a particular group for selecting a particular job on the basis of 4 parameters: Company group level, position offered, pay-scale offered and job location. The following list explains how the categorization was done of the above selected features for studying the candidate behavior.

Candidate Data:

- **Age**: Age was divided into 6 major groups- 20-25, 26-30, 31-35, 36-40, 41-45, <45.
- **Gender**: 2 groups- M or F.
- **Marital Status**: 3 groups- Married (M), Unmarried(U) or Divorcee(D).
- **Education**: 3 groups- Graduation (B), Masters (M) or Doctorate and above (D).
- **Grade**: 3 groups- >80 (High or H), 55-80 (Average or A) and <55 (Low or L)
- **Major**: This was not considered for grouping. This field relates to candidate's validation for a particular job.
- **Experience**: 6 groups- 0 (N), 1-5, 6-10, 11-15, 15-20, >20.
- **Skills**: This was not considered for grouping. This field instead helps in judging the similarity status between the candidate and job. So, only the keywords are considered. Here it is considered that skills are already in place and can be easily represented as vector space model.
- **Location:** 4 groups- North India (N), South India (S), East India (E), West India (W).

Example 2: A candidate having the following details: {27, male, unmarried, graduate, 65%, 2 years experience, New Delhi} can be easily grouped as follows: {26-30, M, U, B, A, 1-5, N}.

Company Data

- **Employer or the Company**: 4 types of company according to the company ranking: Group A (A) (top 25 %), Group B (B), Group C (C), Group D (lowest ranked).
- **Industry field**: Not considered for grouping. Only relates to the major or subject.
- **Position Offered**: 4 groups according to their importance- Level A (Highest Top positions), Level B, Level C and Level D (Initial Level).
- **Pay Scale**: 3 Groups: High (H) (>15 Lakh/ per annum), Medium (M) (6-15 Lakh/ per annum), Low (L) (<=5 L/per annum).

Example 3: A company having the following details: {TCS, Assistant Software Engineer, 4 L/per annum, New Delhi} can be easily grouped as follows: {Group A, D, L, N}.

Domain Knowledge

Extra efforts have to be put in for collecting the domain knowledge regarding the job related fields as well as skills field in the candidate's features list. For the job related fields, domain knowledge regarding the present companies rating need to collected, linked and categorized correctly, for better results. Same is the case with the skills field. As in [3], the hierarchical information regarding the domain is stored; here also the domain knowledge is being stored, for better matching of the skills field.

TABLE I. Sample Matrix Representation of Mined Rules Against Different Job Categories

Age	Gender	Exp	Grade	MStatus	Loc	J1	J2	J3	J4	J5	J6	J7	J8	J9	J10	J11	J12
20-25	-	N	High	-	-	1	1	0	0	1	1	0	0	0	0	0	0
26-30	-	1-5	-	-	-	0	0	0	0	0	1	1	1	0	0	1	1
31-40	M	-	-	-	-	0	0	0	0	1	0	0	0	1	1	1	1
35-40	F	-	-	M	N	1	0	0	0	0	0	1	0	0	1	1	0
35-40	F	-	-	D	N	0	0	0	0	0	0	1	0	0	1	1	0

Example 4: Object Oriented Programming Languages include C++, Java, Python, Smalltalk. So a hierarchical ordering needs to be stored. Refer Fig. 2.
Also, a child here can have more than one parent depending upon whether the child belongs to multiple domains/ groups.

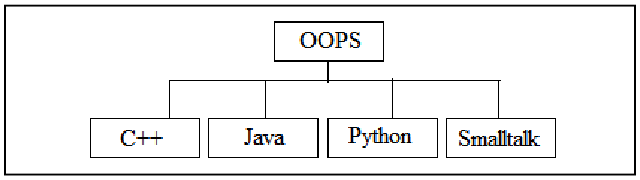

Fig. 2: Hierarchical representation

C. Mining of Decision Tree Induction Rules

Decision trees are tree like graphs or models that represent every possible outcome leading to a particular decision. In these each internal node represents a condition, every edge coming out represents the choice and every leaf node represents the classification or the decision. The path from root node to the leaf represents the decision rule in the form of if-then (Fig. 3). These trees are used in data mining for supervised learning. These are mostly used in classification problems. They are favorable, if the data that is to be tested is categorical. In this, one of the important fact is: selecting the criteria for attribute splitting. Here C4.5 algorithm is used as it uses normalized information gain for attribute selection and splitting. It recursively search for such attributes, that divide the subspaces into highly enriched: one class or the other.

For the candidates, generalization/ categorization was done on the basis of various features as discussed earlier. For the jobs, they were divided into total 20 different meaningful categories, each having unique combination/ characteristic in terms of company groups, position level, pay-scale and location. The data was collected and analyzed, for each job category, separately. Exhaustive search was made for all the rules that can determine the choice of candidates job taste, belonging to a particular group, for the corresponding job category. The evaluation criterion taken for determining the strength of these rules are the lift, confidence and the sample size [10]. Lift determines the importance of a rule and confidence represents the reliability of a rule. Lift and confidence are calculated as follows:

$$\text{Lift A (Rule i)} = \frac{\text{P (target class A| subset i)}}{\text{P (target class A| population)}} \quad (1)$$

$$\text{Confidence A (Rule i)} = \text{P (class A| subset data by Rule i)} \quad (2)$$

Here, the lift value was considered to be greater than 1 and confidence percentage to be greater than 80%. A threshold value for the sample size was taken. If there exists a rule which has high lift and confidence but its sample size is less than the threshold value, then it is not taken into consideration, however strong the rule may be. All the rules that crossed the selection criteria were enlisted and checked for redundancy. After that, a common matrix representing all the job categories preferences, for a particular rule was made. In this matrix, corresponding to a particular rule, if the rule exists for a particular job category, then the corresponding field of the job category is made 1 else 0. Refer Table 1.

$$\text{Matrix [i, j]} = 1 \quad (\text{if } i^{th} \text{ rule exists for } j^{th} \text{ job category})$$
$$0 \quad (\text{else } 0) \quad (3)$$

Example 5: In Table 1, for representation, there are 5 rules and 12 job categories. 1st rule indicates that for the people belonging to age-group 20-25 and having experience = null and grades = high, often selected the jobs belonging to the category J1, J2, J5 and J6.

After this 4 preference matrix were generated, each one preserving the authenticity of generated rules. And the scores/ weights assigned were normalized by dividing them with the total number of instances available for the jobs i.e. respective probabilities of selecting a particular field are stored.

Example 6: Preference matrix for company stores information regarding its 4 type of company's preference. 1st position for Group A companies preference, 2nd for Group B, 3rd for Group C and 4th for Group D as: [p_A p_B p_C p_D].

Example 7: Last Rule (Table 1), Age=35-40, gender=Female and MStatus=Divorcee. The 4 preference matrices are Company Matrix: [0 1/3 2/3 0], Position Matrix: [0 2/3 1/3 0], Income Matrix: [2/3 1/3 0], Location Matrix: [1/1 0 0 0].

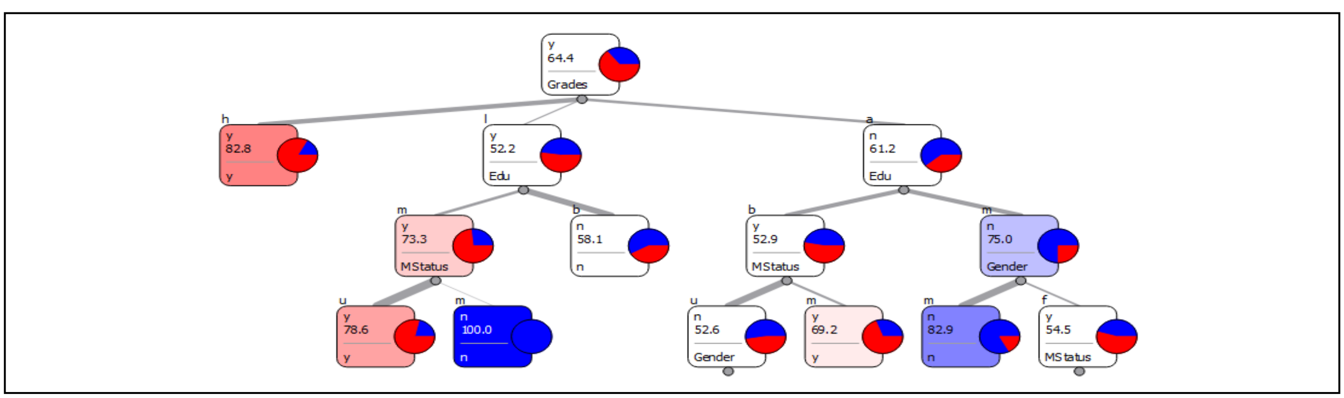

Fig. 3. Pictorial representation of a decision tree induction rule

D. Recommendations Generation

Let's understand the complete procedure with the help of an example:

Example 8: Here is a candidate that has the following details: {27, male, unmarried, graduate, Computer Science, 65%, 2 years experience, New Delhi, (C++, Oracle, Data Structures, Algorithms, Machine Learning, English)} and we have the following 10 jobs in our database: {J1, J2, J3, J4, J5, J6, J7, J8, J9, J10}. Refer to Table 2.

TABLE II. SAMPLE JOBS IN THE DATABASE

Jobs	Field	Qualification	Min. Experience
J1	CSE	Graduation	2
J2	CSE	Graduation	2
J3	M.E.	Graduation	2
J4	CSE	Graduation	1
J5	CSE	Graduation	4
J6	CSE	Graduation	1
J7	CSE	Graduation	0
J8	CSE	Graduation	2
J9	CSE	Graduation	3
J10	CSE	Masters	1

Phase 1

Step 1: *Shortlist the jobs for which the candidate is currently eligible for:* The fields considered for short listing are: major, min. Qualification required and min. experience needed for the job. So, from the above jobs list: J3, J5, J9 and J10 get eliminated. Hence we are left with only 6 jobs in our basket {J1, J2, J4, J6, J7 and J8}. Also here remove the redundancy in jobs, if any.

TABLE III. SAMPLE CALCULATION OF COSINE SIMILARITY

Job	Skills	Skills Vector	Candidate Vector	Cosine similarity
J1	MySQL, PHP, Data structure, English.	[1,1,1,1]	[0,0,1,1]	.707
J2	OpenCL, Networking, English, German	[1,1,1,1]	[0,0,1,0]	.5
J4	Python, MySQL, Machine Learning, English, German	[1,1,1,1,1]	[1,0,1,1,0]	.774
J6	C++, Data Structure, Algorithm, English	[1,1,1,1]	[1,1,1,1]	1
J7	C++, Oracle, Algorithms, English.	[1,1,1,1]	[1,1,1,1]	1
J8	Smalltalk, DB2, JSP, Machine Learning, English	[1,1,1,1,1]	[1,0,0,1,1]	.774

Step 2: *Calculate the Content Based Similarity:* Now calculate the similarity index for the short listed jobs with respect to the candidate. The similarity index is calculated in between the jobs desired skills field and candidate's possessed skill's fields. Here cosine similarity between the two is considered. Cosine similarity is between 2 vectors a & b is calculated by the following formula:

$$\text{Cosine similarity } (a, b) = (a \cdot b) / (|a|\ |b|) \tag{4}$$

Firstly preferred matrix vector for job's skills requirement is created and then accordingly candidate's vector is created [2]. However, if a level of proficiency is required in a particular language, than it can be represented as the corresponding weight in the vector. Example if the requirement is to have a proficiency level of 3 on a 5 scale, in java language then that can be represented with the weight 3 in the corresponding java field representation in the job vector. Also, if 2 languages/ skills match exactly or are related hierarchically and belong to same domain, then that is considered as an exact match and weight 1 is assigned else a 0 is assigned [3]. Refer to Table 3.

Step 3: *Apply the Decision Tree Induction Rules for the category to which the candidate belongs:* Here, the basic categorization of jobs is done firstly. After that these categories are matched according to the preference matrices of the generated rules and assigned preference weights accordingly. So, for the above mentioned candidate: {26-30, M, U, B, A, 1-5, N}, the preference matrices for a candidate belonging to age group: 26-30 and having experience: 1-5 years, are: company [0,1/5,2/5,2/5], position [0,2/5,3/5,0] and pay-scale [1/5, 3/5, 1/5]. Location preference matrix is not considered as it is assumed that the group has applied equally in all 4 regions and hence adding these will not result into any new information. Now, assign proper weights to the corresponding jobs, for building up accuracy in recommendations, by judging the general behavior of the respective group candidates. Normalize the rule's weight with the help of following equation:

$$\text{Normalized Weight} = (W_i - W_{min}) / (W_{max} - W_{min}) \tag{5}$$

Where W_i is the rule's weight assigned to the corresponding i^{th} job and W_{min} and W_{max} represent the min and max weights considering all the job's weight altogether.

Step 4: *Generate the final weights:* Now calculate the final weight score by summing up values of cosine similarity and rules weight according to the following equation:

$$\text{Final Weight } (i) = w_1 \cdot sim_i + w_2 \cdot rw_i \tag{6}$$

Where i represents ith shortlisted job, sim_i stands for cosine similarity for ith job, rw_i represent rules weight assigned to the ith job, w_1 and w_2 represent the weights assigned to cosine similarity and rules weight for preserving their relevance. Here w_1 and w_2 both have the value as .5.

Step 5: Sort *the jobs in descending order*: According to the final score obtained, sort the jobs in descending order. The final recommendations provided to the candidate (Example 8) are {J6, J4, J8, J1, J2, J7} as shown in Table 4.

Phase 2

This phase starts when the candidate has applied for at least 10 jobs. These jobs further fulfill the concept of mining new information, for recommending new jobs to the respective candidate, according to his/her job taste. The minimum jobs for rules creation is 10 and maximum jobs considered at a time were Θ where $30 < \Theta < 40$ as after Θ no accurate generalized results were obtained.

TABLE IV. SAMPLE CALCULATION SHOWING FINAL RANKING OF JOBS AFTER PHASE 1

Job	Cosine similarity	Company	Position	Pay Scale	Rules Weight	Normalized Rules Weight	Final Weight	Final Ranking
J1	.707	B	C	H	{1/5+3/5+1/5} = 1	.4	.5535	4
J2	.5	B	B	M	{1/5+2/5+3/5} = 1.2	.6	.55	5
J4	.774	D	B	M	{2/5+2/5+3/5} = 1.4	.8	.787	2
J6	1	C	C	M	{2/5+3/5+3/5} = 1.6	1	1	1
J7	1	D	D	L	{2/5+0+1/5} = .6	0	.5	6
J8	.774	C	C	L	{2/5+3/5+1/5} = 1.2	.6	.687	3

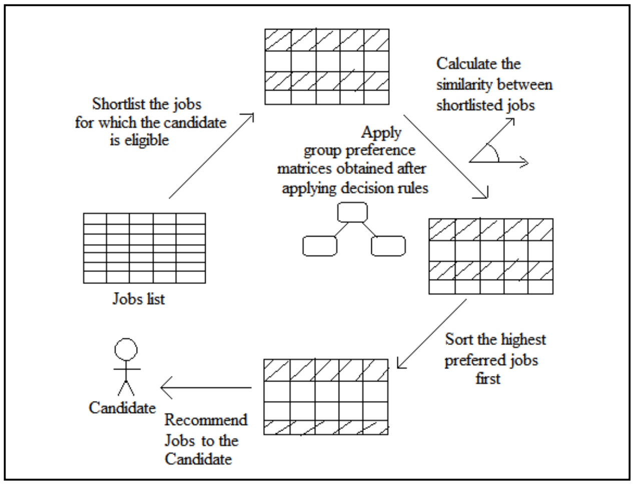

Fig. 4. Phase 1 of job recommendations

Step 1: *Shortlist the jobs for which the candidate is currently eligible for:* This step is same as earlier. Here also first the short-listing of jobs is done firstly, for working efficiently on the small jobs set, for which the candidate is currently eligible for. Considering the Example 8, we are again left with only 6 jobs in our basket or job set: {J1, J2, J4, J6, J7 and J8}.

Step 2: *Calculate the Content Based Similarity:* Calculate the similarity index between the job's required skills and candidate possessed skills using equation (4). So according to the job vector similarity, cosine similarity index is created for each job. For Example 8, it is same as calculated in Table 3.

Step 3: *Direct preference matrix creation according to the customer's latest preferences for jobs:* This step is different from the first phase step third. Here preference matrices are created by directly judging candidate criteria behind selecting a job. The probabilities are calculated directly by keeping track of the company group, level of position offered, pay-scale applied for and the location of the job. Suppose we have the list of following 10 jobs that the customer has recently applied for: {J11,J12,J13,J14,J15,J16,J17,J18,J19,J20} and from these jobs, suppose the preference matrices are as follows: Company [1/10 2/10 5/10 2/10] Position: [0 6/10 3/10 1/10] Pay-scale: [2/10 8/10 0] Location: [7/10 1/10 0 2/10]. Again assign proper weights to the corresponding jobs, for building up accuracy in recommendations, by judging the personalized behavior of the candidate. After that, normalize the rule's weights by using the normalization equation (5).

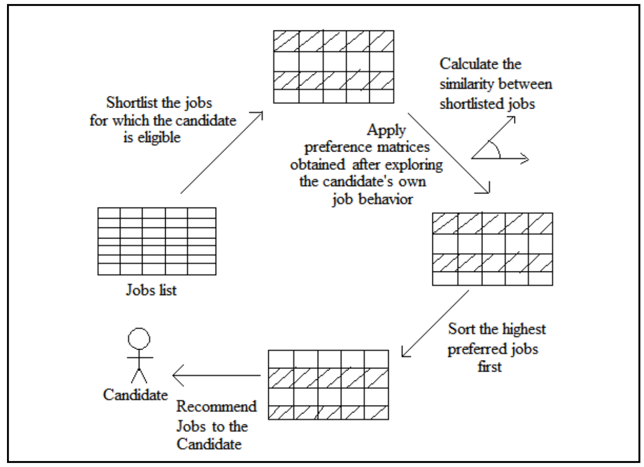

Fig. 5. Phase 2 of job recommendations

Step 4: *Generate the final weights:* As in earlier phase, calculate the final weight score using equation (6).

Step 5: *Sort the jobs in descending order:* Sort the jobs in descending order according to the final score and recommend them to the customer. Final personalized list of Job recommendations for candidate (Example 8) after knowing his job preferences are: {J6, J4, J7, J8, J2, J1}. Refer to Table 5.

Fig. 4. and Fig. 5. represent both the phases of recommendations.

V. RESULTS AND DISCUSSIONS

In the research dataset, there were 1500 candidates and 500 jobs. 20 different meaningful categories of jobs were made and all the jobs were categorized into these categories. Python was used as implementation language for making recommendation as well as data mining. The Orange library [11] was used for data mining purpose. The relevant rules were mined against each job category and stored as knowledge base for creating further recommendations.

TABLE V. SAMPLE CALCULATIONS SHOWING FINAL RANKING OF JOBS AFTER PHASE 2

Job	Cosine Similarity	Company	Position	Pay Scale	Loc	Rules Weight	Normalized Rules Weight	Final Weight	Final Ranking
J1	.707	B	C	H	W	{2/10+3/10+2/10+2/10}=.9	.222	.4645	6
J2	.5	B	B	M	S	{2/10+6/10+8/10+1/10}=1.7	.6666	.5833	5
J4	.774	D	B	M	N	{2/10+6/10+8/10+7/10}=2.3	1	.8887	2
J6	1	C	C	M	N	{5/10+3/10+8/10+7/10}=2.3	1	1	1
J7	1	D	D	L	W	{2/10+1/10+0+2/10}=.5	0	.75	3
J8	.774	C	C	L	N	{5/10+3/10+0+7/10}=1.5	.5555	.6645	4

TABLE VI. COMPARISON BETWEEN THE DIFFERENT JOB RECOMMENDER SYSTEMS

	CASPER [3]	PROACTIVE [4]	BILATERAL [5]	iHR [7]	Machine Learned [8]	Proposed System
Input For User Profile	Candidate Behavior and Search Query	Candidate Information	Candidate Information	Candidate Information and Behavior	Candidate Information	Candidate Information and Behavior
Recommendation Strategy	Memory based & Cluster based CFR, Case Based Reasoning	CBR, KBR	Probabilistic Hybrid Recommendation Engine	Different strategies for different groups (CBR, CFR, HyR)	Supervised learning Based DTNB Hybrid Classifier	CBR, Model based CFR, KBR
Output of Recommendation	One complete list of recommended jobs	4 different job recommendation lists	List of recommended jobs as well as list of recommended CVs	One complete list of recommended jobs	List of predictions of next job positions	One complete list of recommended jobs
User Feedback Mechanism	Implicit	Explicit	Explicit	Implicit	-	Implicit
Pros	1. Adapts to user needs 2. Implicit user preference considered	1. Four different recommendation modules 2. Uses ontology to cluster jobs	1. Two sided recommendations	1.Different recommendation strategies for different clusters of users	1.Considers transition history for recommendations	1. Implicit user preference considered 2. Adapts to the user needs 3. Different recommendations for different group of users
Cons	1. User profile not much accurate 2. One way recommendation	1. One Way Recommendation 2. Explicit Feedback required 3.Knowledge engineering problem	1. Explicit Feedback Required 2. There are no perfect or standard methods/ measures defined	1. One Way Recommendation 2. Data Sparsity problem	1. One Way Recommendation 2. Data Sparsity problem	1. One Way Recommendation 2. Data Sparsity problem at initial level

The prediction accuracy was used as a measure to judge the importance of made recommendations [12]. CBRS stands for Content based recommender system, CFRS for collaborative filtering based recommender systems, PRS-I and PRS-II for our Proposed Recommender System Phase I and Phase II, respectively.

Fig. 6. and Fig. 7. shows the graphs which compare the prediction accuracies of content based, collaborative filtering and proposed system approaches for phase I and phase II, in percentage, for Top N Recommendations made to the candidates.

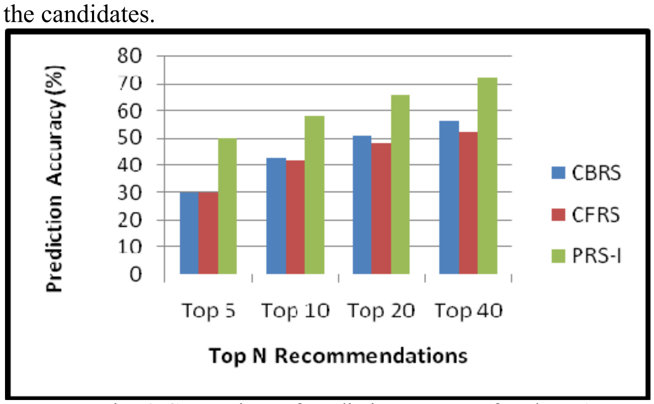

Fig. 6. Comparison of prediction accuracy for phase 1

In first phase, a prediction accuracy of about 72 percent was perceived as against lower percentages of traditional recommender methods. However, in second phase, a significant prediction accuracy of about 80 percent was attained.

The reason behind the difference between the prediction accuracies of 2 phases lies in the fact that in first phase the filtering is based on content based filtering and the generalized rules that exist in the knowledge base for that particular candidate group.

Whereas, in the second phase the recommendations just are more candidate/ customer centric as those depend on the latest job preferences of the candidate. It also proves to be efficient as it eliminates the irrelevant, outdated and stale jobs out of the basket and considers only latest or jobs that are freshly (according to the time period) applied by the candidate.

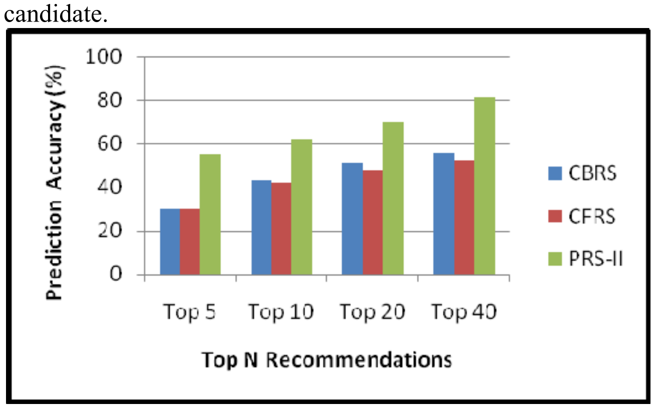

Fig. 7. Comparison of prediction accuracy of for phase 2

Table 6 compares the different research based recommender systems with that of the proposed system. Here, CBR stands for Content Based Recommender, CFR for

Collaborative Filtering Recommender, HyR for Hybrid recommender and KBR for Knowledge Based Recommender.

Now concentrating on the limitations, one is that while categorizing the company data on the basis of job positions, job ranking (levels) and pay-scale offered, there were certain trade-offs. For an example, considering 2 companies, one having level A in the job market, but offering a low package for a high position for its job. Whereas the other also of level A in the job market, but offering a high package for the same position for its job. So, here although the positions are same, company levels are same but still the packages are categorized into 2 different categories. This leads to a mismatch, unevenness of the normal trend in the existing categorization system. So, such trade-offs were made, in the data categorization part of the existing system.

Another limitation lies in the first phase such that, if no rule in the knowledge base matches the candidate's group or category, then no preference matrices are generated and as a result, the algorithm of first phase merely reduces to content based recommender system, in worst case. However, as the candidate applies for a minimum of 10 jobs, the second phase starts over and overcome this limitation regarding the job preferences and efficiency is thus re-gained.

VI. CONCLUSION

In this paper, the efforts were put to take into consideration the job preferences of the candidates along with the content based profile matching. It along with increasing the prediction accuracy also helped to solve the problem of providing direction to the candidates who are not clear about their job goals as general group preferences are imposed. However, in case of candidates having exceptional path carriers, the system adapts itself by focusing on their latest job preference behavior and providing them the list of recommendations accordingly. Tracking the present preferences of the candidate regarding the job, helps to prioritize only the relevant jobs as against the irrelevant jobs that are shortlisted after the content based matching of the candidate.

REFERENCES

[1] S. Zheng, W. Hong, N. Zhang, F. Yang, "Job Recommender Systems: A Survey", 7th International Conference on Computer Science & Education (ICCSE), Melbourne, VIC, pp. 920 – 924, 14-17 Jul. 2012.

[2] S. T. Al-Otaibi, M. Ykhlef, "A survey of Job Recommender Systems", International Journal of the Physical Sciences, Vol. 7 (29), pp. 5127-5142, 26 Jul. 2012.

[3] R. Rafter, K. Bradley, B. Smyth, "Personalized Retrieval for Online Recruitment Services", In: Proceedings of the 22nd Annual Colloquium on Information Retrieval(IRSG 2000), Cambridge, UK, 5-7 April, 2000.

[4] D. H. Lee, P. Brusilovsky,"Fighting Information Overflow with Personalized Comprehensive Information Access: A Proactive Job Recommender", Third International Conference on Automatic and Autonomous Systems, ICAS07, Athens, pp. 21, 19-25 Jun. 2007.

[5] J. Malinowski, T. Keim, O. Wendt, T. Weitzel, "Matching People and Jobs: A Bilateral Recommendation Approach", Proceedings of the 39th Annual Hawaii International Conference on System Sciences, HICSS '06, vol 6, pp. 137c, 04-07 Jan. 2006.

[6] H. Yu, C. Liu ,F. Zhang, "Reciprocal Recommendation Algorithm for the Field of Recruitment", Journal of Information & Computational Science, vol 8 (16), pp. 4061–4068, 2011.

[7] W. Hong, S. Zheng, H. Wang, J. Shi, "A Job Recommender System Based on User Clustering", Journal of Computers, vol. 8(8), 1 Aug. 2013.

[8] Paparrizos, I., Cambazoglu, B. B., Gionis, A., "Machine learned Job Recommendation", Proceedings of the fifth ACM conference on Recommender systems, RecSys '11, ACM, New York, NY, USA, pp. 325-328, 23-Oct. 2011.

[9] W. Hong, S. Zheng, H. Wang, "Dynamic User Profile-Based Job Recommender System", 8th International Conference on Computer Science & Education (ICCSE), Colombo, pp. 1499-1503, 26-28 Apr. 2013.

[10] C. F. Chien, L. F. Chen, 2008, "Data Mining to improve personnel selection and enhance human capital: A case study in high-technology industry", Expert Systems with Applications: An International Journal, vol 34(1), pp. 280-290, Jan 2008.

[11] Demšar, J., Zupan, B., "Orange: Data Mining Fruitful and Fun - A Historical Perspective", Informatica Slovenia, vol 37(1), pp. 55–60, 2013.

[12] F. H. d. Olmo, E. Gaudioso, "Evaluation of recommender systems: A new approach," Expert System with Applications, vol 35, pp. 790-804, 2008.