Algorithms and Optimization of Big Data End–Semester Exam

Jay A. Joshi (1401005)
Ahmedabad University, jay.jo.btechi14@ahduni.edu.in

***Abstract* – Data mining is the computing process of discovering patterns in large data sets involving methods at intersection of artificial intelligence, machine learning, statistics and database systems. Many applications are trending in current scenario and a recommendation system is one of them. Using Novel Filtering techniques, project aims to create a recommendation system that can suggest career progression path to its registered users.**

Index Terms - ***Data mining, Data cleansing, recommendation systems***

INTRODUCTION

Recommendation system can be assimilated as a general data mining problem. Approach to the problem can be divided into 3 parts : Data Preparation, Data Mining and Post Processing. Data preparation includes Feature selection, dimensionality reduction, and normalization. Data mining involves tasks of anomaly detection, dependency modeling, clustering, classification, Regression and summarization. Post processing does the job of data validation, pattern interpretation and data visualization.

RECOMMENDATION SYSTEMS

Recommendation methods can be classified into few types as follows :-

- **Collaborative filtering:** Collaborative filters are based on collecting and analyzing a large amount of data of information on user's behavior, activities and preferences and predicting what user will like based on based on similarity with other users.

  Facebook, LinkedIn and other social networks use collaborative filtering to recommend new friends, groups, and other social connections by examining the network of connections between a user and their friends.

  Algorithms like K-NN and Pearson's Correlation are used to measure the similarity. This method produces good enough results with minimum requirements. It can be further classified into personalized and non-personalized collaborative filtering.

- **Content-based filtering:** Content based filtering methods are based on a description of the item and a profile of the user's preference. Here, keywords are used to describe the items and a user profile is built in indicate the type of item this user likes. Direct feedback from user can be used to assign weights to attributes. This allows recommendation of new and unpopular item. This filtering method is used by Rotten tomatoes to rate movies.

  This technique may use NLP to extract content features. This filtering makes it difficult to implement serendipity.

- **Novel methods:** Unlike traditional methods of collaborative and content filtering, this technique learns to rank. Ranking function minimizes the loss function defined on individual relevance judgment. Ranking score is based on regression and classification. Regression techniques like ordinal regression, Logistic regression, Support Vector Machine algorithm etc. are used. It becomes one of the most efficient way to create personalized recommendation system by using neural network and deep learning approaches.

- **Hybrid approaches:** It combines many approaches like collaborative filtering, content-based filtering , demographic and knowledge based filtering technique. This helps overcome limitations of each system. Netflix is the best example of hybrid recommendation system. It uses naïve Bayes and k-NN classifiers.

SYSTEM OVERVIEW

Data set: Candidate Profile Data set

Given data set consists raw data of 39 different job profile for which the algorithm will help create a career path recommendation. Raw data needs to be pre-processed for further data mining processes.

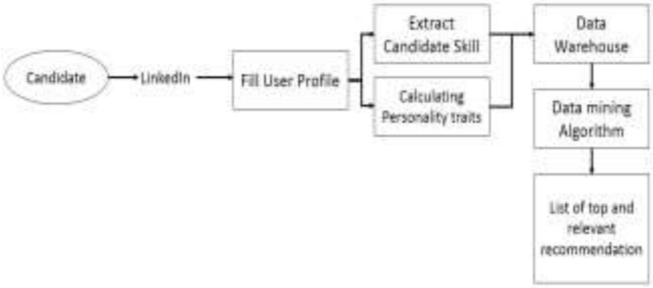
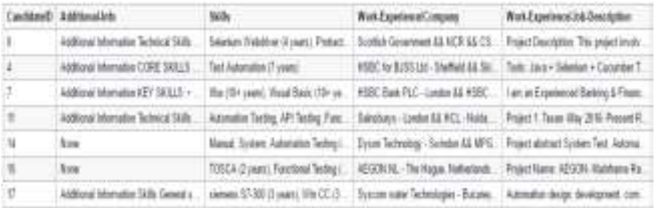


Figure:1 System Architecture

DATA CLEANSING (STEP 1): The process includes assembling data under correct header, removing corrupt inaccurate records either via strict validation or fuzzy validation. It also includes standardization of data. Here, data is cleaned using online tool to obtain a .csv file. This file contains the database with standardized headers like Candidate ID, Skills, Work experience, education etc.in well sorted way to perform further operations.



CandidateID	AdditionalInfo	Skills	WorkExperienceCompany	WorkExperienceJobDescription
1	Additional Information Technical Skills ...	Selenium Webdriver (4 years), Protract...	Scottish Government && NCR && CS...	Project Description: This project involv...
4	Additional Information CORE SKILLS ...	Test Automation (7 years)	HSBC for BJSS Ltd - Sheffield && Sh...	Tools: Java + Selenium + Cucumber T...
7	Additional Information KEY SKILLS • ...	Vba (10+ years), Visual Basic (10+ ye...	HSBC Bank PLC - London && HSBC ...	I am an Experienced Banking & Financ...
11	Additional Information Technical Skills ...	Automation Testing, API Testing ,Func...	Sainsburys - London && HCL - Noida ...	Project 1: Team May 2016 - Present R...
14	None	Manual, System, Automation Testing (...	Dyson Technology - Swindon && MPS ...	Project abstract System Test, Automa...
16	None	TOSCA (2 years), Functional Testing (...	AEGON NL - The Hague, Netherlands ...	Project Name: AEGON- Mathhams Ra...
17	Additional Information Skills General ...	siemens S7-300 (3 years), Win CC (3...	Syscom water Technologies - Bucures...	Automation design, development, com...

Figure:2 Data Cleaning

LEARNING TO RANK (STEP 2)

In this Novel Filtering technique, learning algorithms are applied to solve the candidate ranking problem in recommendation systems. In the ranking problem, a scoring function h(x) outputs the relevance score, which reflects how well a candidate profile fits the requirements of a given job position. As the relevance score is a continuous variable, the candidate ranking problem can be reduced to a regression problem where the candidate scoring function must be learned using supervised learning techniques. Then the system outputs the final ranked list by applying the learned function to sort the candidates.

The score function h(x) derives the candidate's relevance degree y_i from the values of his feature vector x_i. The feature vector x_i consists of a set of m attributes $\{a_1, \ldots, a_m\}$ that correspond to the candidate's skill criteria. These can be either continuous variables (representing a candidate's feature assessed on numerical scale) or Boolean variable(whether he has that skill or not).The true scoring function is usually unknown and an approximation is learned from the training set D. In the proposed system, we consider that the training set consists of a set of N previous candidate selection examples, given as an input to the system:

$$D = \{(x_i, y_i | \, x_i \in R^m, y_i \in R\}_{i=1}^{N}$$

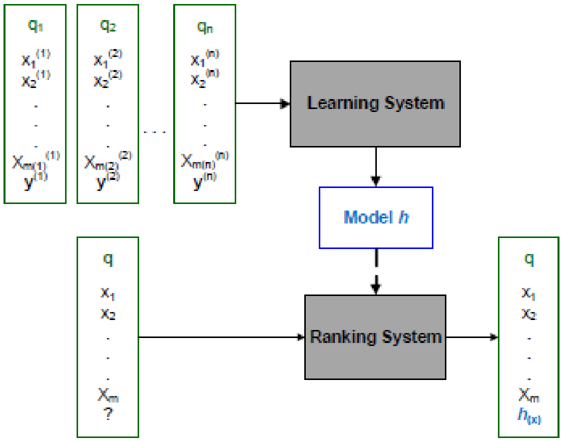


Figure:3 Learning to Rank Model

LINEAR REGRESSION (STEP 3)

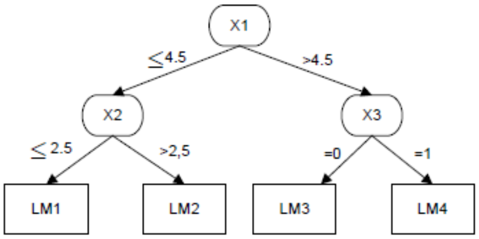


Figure:3 Regression Tree with Linear Models of Regression

In linear regression, the relevance score y_i of the i_{th} candidate is predicted as a linear function of the recommendation criteria, which comprise the candidate's feature vector x_i plus noise e (regression error):

$$y_i = w^T x_i + e$$

The linear regression algorithm finds the optimal parameter vector w that minimizes the regression error.

SUPPORT VECTOR REGRESSION (STEP 4)

Support Vector Machines (SVMs) are a set of related methods for supervised learning. SVMs comes from the kernel representation, which allows a non-linear mapping of input space to a higher dimensional feature space. The objective of Support Vector Regression is to find a function f that minimizes the expected error – i.e., the integral of a certain loss function – according to the unknown probability distribution of the data. This minimizes the risk that the estimated function differs from the original. Assuming N data points and a Kernel K, the support vectors and the support values of the solution define the following regression function:

$$f(x) = \sum_{i=1}^{N} a_i K(x, x_i) + b \mid b, a_i \in R$$

ALGORITHM

1. Extract the concepts from User Profile Columns: "Skill-Set", "Experience", "Area of Interest".
2. Store it in an associative array C_{UP} with key as concept-id and value as term frequency of occurrence.
3. Extract the concepts from user's Follow list and store it in an array C_{FL} with corresponding term frequency.
4. Extract the concepts from user's navigation list and store it in an array C_{NL} with corresponding term frequency.
5. Combine the array in one array with different weightages. $C = 3*C_{UP} + 2*C_{FL} + 0.25*C_{NL}$
6. Expand the concept-list C based on co-occurrence table and update the frequency as well. C_{EX} = C U (Concepts co-occurring with C).
7. Create Weight vector of length 'n', where n is the total no of concepts in C_{EX}. = [wi] where wi is the weightage or term frequency of concepts within C_{EX}
8. Rank the relevant skills with constantly learning technique.
9. For each resource calculate similarity with Support Vector Machine (SVM) representing the user interest as:
10. Arrange the resources in decreasing order of similarity value.
11. Filter out top 'u' recommend any 3 jobs out of them.

.

REFERENCES

[1] XavierAmatriain," Introduction to a recommender system – A 4 hour lecture"http://technocalifornia.blogspot.in/2014/08/introduction-to-recommender-systems-4.htmlAccessed: April 25, 2017.

[2] Giannis Tzimas, Evanthia Faliagka. "Application of machine learning algorithm to online recruitment system" http://s3.amazonaws.com/academia.edu.documents/38834077/csit3648.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1493296357&Signature=3GrtP5pSvQ4OiUo71DCig8VpbQo%3D&response-content-disposition=inline%3B%20filename%3DCONTEXTUAL_MODEL_OF_RECOMMENDING_RESOURC.pdf Accessed: April 24,2017

[3] Anoop kumar Pandey, Amit Kumar, Balaji Rajendran https://www.researchgate.net/profile/Giannis_Tzimas/publication/265882207_Application_of_Machine_Learning_Algorithms_to_an_online_Recruitment_System/links/54b7c6630cf2c27adc4716a5.pdf Accessed:April26,2017.