Skills Recommendation System for LinkedIn

Ratnesh Shah - 1401110
School Of Engineering & Applied Sciences, Ahmedabad University.

***Abstract*—During the last decade the amount of information available online increased exponentially. Recommender system address the problem of filtering information that is likely of interest to individual users. Skill recommendation system can be very useful to the people who want to boost up their career profile. Using the data available from the skills of people who are quite successful and experienced in their profession we can guide the people in their career choices. This problem is one the key topics of research under the domain of recommendation systems.**

Keywords*—*Skill Recommendation, Weighted Matrix, Cosine Similarity, Profession Recommendation

I. Motivation

Making the right choice for career is one of the most difficult task which most student of undergraduate students face. I myself have faced this problem at point of time. Normally what people do at this point of time is they go for consulting some experienced people to get a proper guidance and paying them huge fees. But the question to ponder is we have so many data of people on websites such as linkedin, Xing and Naukri etc. which can give us a better suggestion and so I have tried to build one such system to help students like me or even professionals to acheive their career goal.

II. Introduction

Recommender System (RS) is a class of applications dealing with information overload. As more and more information is published on the World Wide Web, it is difficult to find needed information quickly and efficiently. RS helps solve this problem by recommending items to users based on their previous preferences. Many applications have used recommender systems, especially in the e-commerce domains. Typically in a recommender system, there is a set of users and a set of items. Each user u rates a set of items by some values. The task of a recommender system is to predict the rating of user u on an unrated item i or recommend some items for user u based on the existing ratings. Techniques in RS [Adomavicius and Tuzhilin, 2005] can be divided into three categories: collaborative filtering, content-based recommendation and hybrid approaches. Collaborative filtering make recommendations based on the ratings of item i by the set of users whose rating profiles are most similar to that of user u. Content-based methods use the features of items, e.g. movies genres, directors, actors, etc., to generate recommendations. Hybrid approaches [Burke, 2002] make recommendations by combining collaborative filtering and content-based recommendation.

III. Model Description

The first step I performed was data cleaning. The data provided to us has skills which are separated by commas,hyphens, fullstops, bullets etc. so to get the list of all the skills of all the profession I cleaned it to get the desired output. Next thing I extracted the candidate id, job title and the skills information from all files and made a common file containing these information. Also I separated the data of candidate id and the skills he/she possess. So finally this three files are used in both the modules explained below.

A. Module-1

Statement: A module that reads users profile and suggest a career path in terms of skillset to be acquired.
The method I have implemented comes under collaborative filtering technique. The complete method can be visualized from the below figure. We consider a set $U = \{U_1, U_2, U_3, ...\}$

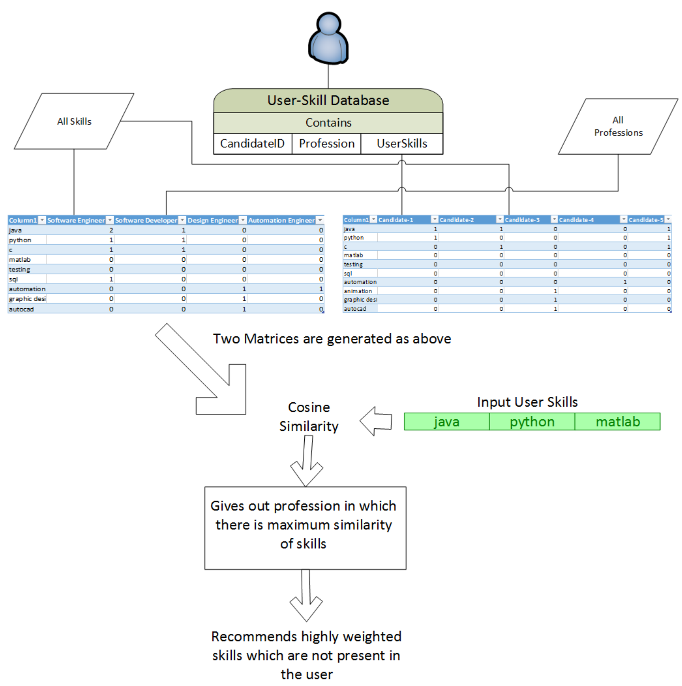

Fig. 1: Implemented Model

of user profiles, each of them being the description of a specific person. To each profile u is associated a set S(u) of skills, representing the skill-set which the corresponding person declares to have. The same skill may be associated to

more than one users. The set of all distinct skills is denoted by $S = \{S_1, S_2, S_3, ...\}$. To each profile u is also associated a current job position p(u). The same job position may be the current one for more than one profile. We denote by $P = \{P_1, P_2, P_3, ...\}$ the set of all distinct job positions.
We extract a vector based representation, where skills are used as features. Specically, from the set U of known profiles, we build a S x P matrix C counting the co-occurrences between skills and positions across them. It is a weighted matrix in which the weight is assigned as per the no. of users of that job position having that particular field. The reason for making it a weighted matrix is that with the help of weight the preference or we can say the importance of that particular skill which if not found in user than recommended first.
Next, the second matrix I construct is S x U which is a binary matrix containing 1 if the user has that particular skill else there is 0.

$$UserSkillMatrix_{ij} = \begin{cases} 1, & \text{if } S_i = S(U_j) \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

The constructed matrices can be visualized from the above model diagram.

Now, as an input a user inputs the skills which he already possess then using this input skills we create a vector of skills from S where there is a 1 if the input skill matches with the skill from S.
The next and the most important step is performing cosine similarity between this input skill binary vector and the C matrix.
As an output of cosine similarity what we get is first k job profile for which the input user skills are matching the most. Now, we have in the matrix C the data of all the skills taken by the people in that profession along with its weight. The skill set is sorted in descending order of their weights. If the skill is not present in the input user skills then that skill is recommended. Thus we perform recommendation of skills as per importance of that skill.

Cosine Similarity:
The cosine similarity between two vectors (or two documents on the Vector Space) is a measure that calculates the cosine of the angle between them. This metric is a measurement of orientation and not magnitude, it can be seen as a comparison between documents on a normalized space because were not taking into the consideration only the magnitude of each word count (tf-idf) of each document, but the angle between the documents. What we have to do to build the cosine similarity equation is to solve the equation of the dot product for the $\cos(\theta) = \frac{\boldsymbol{a} \cdot \boldsymbol{b}}{||\boldsymbol{a}|| \cdot ||\boldsymbol{b}||}$

Why do we use cosine similarity instead of euclidean distance?
Suppose we have a document with the word sky appearing 200 times and another document with the word sky appearing 50, the Euclidean distance between them will be higher but the angle will still be small because they are pointing to the same direction, which is what matters when we are comparing documents. Same thing goes for comparing skills.

B. Module-2

Statement: A module in which user enters a career goal and based on this career goal and other related information the platform suggest a career path.

The complete model of module-2 can be visualized from below figure:

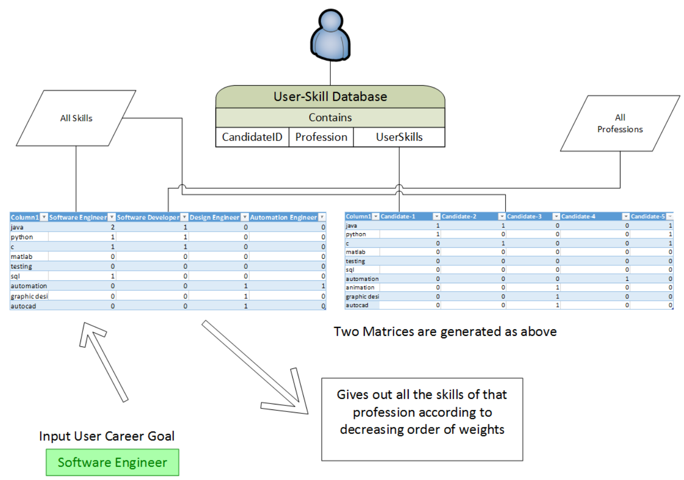

Fig. 2: Recommends skills for a career

User give his target career goal(profession) as input. Find index of target profession from positions vector P using string matching. From obtained index extract column vectors at that index position in C matrix.So we obtained weighted skill vector of target profession. Sort skills in descending order of weights. Recommend skills from higher weight to lower weight for targeted profession.

Thus, given input a career goal a person can get the list of skills he/she should acquire.

IV. Results

A. Module-1

User Input of skills is given: java,python

As we can see that the user already has java and python skills. So when user inputs this skills a binary vector is created having 1's at indexes of which have java and python as skills. Cosine Similarity is applied on this vector and the C vector generated. The result of which is a vector containing values between 0 and 1 for each column, where 1 shows complete matching of input skills to skill vector of a profession and 0 shows no skill matches. From this we know that the max of the cosine correlation vector gives us the column no. of the Profession with maximum skill match. Hence we recommend that particular profession.

```
c =

     2     1     0     0
     1     1     0     0
     1     1     0     0
     0     0     0     0
     0     0     0     0
     1     0     0     0
     0     0     0     1
     0     0     1     0
     0     0     1     0
     0     0     1     0
```

Fig. 3: C matrix(S x P)

```
user_skill_mat =

     1     1     0     0     1
     1     0     0     0     1
     0     1     0     0     1
     0     0     0     0     0
     0     0     0     0     0
     0     1     0     0     0
     0     0     0     1     0
     0     0     1     0     0
     0     0     1     0     0
     0     0     1     0     0
```

Fig. 4: (S x U) Matrix

```
Skill Recommendation System: Ratnesh Shah - 1401110
Please enter your skills: java,python
1st preference -->You are recommended to pursue software developer profession
You are recommend to acquire following skills-->
    'c'

-----------------------------------------------------------------
2nd preference -->You are recommended to pursue software engineer profession
You are recommended to acquire following skills-->
    'c'

    'sql'
```

Fig. 5: Results of Module-1

B. Module-2

The user input is a profession: Design Engineer/ Software Engineer

```
Skill Recommendation System: Ratnesh Shah - 1401110
Enter your profession: design engineer
Recommended skills:(Ordered from High Preferance to Low Preference)
    'graphic design'    'animation'    'autocad'
```

Fig. 6: Recommended skills for Design Engineer

V. Proof of Correctness

We have a labelled data which contains the user and the skills possessed and the profession he/she belongs to. So if I give all the skills a user which is already in the data set than the recommended profession should match with the actual profession of the user.

```
Skill Recommendation System: Ratnesh Shah - 1401110
Enter your profession: software engineer
Recommended skills:(Ordered from High Preferance to Low Preference)
    'java'    'python'    'c'    'sql'
```

Fig. 7: Recommended skills for software Engineer

1	software engineer	java,python
2	software engineer	java,c,sql
3	design engineer	graphic design,animation,autocad
4	automation engineer	automation
5	software developer	c,python,java
6	design engineer	graphic design

Fig. 8: Data Present in the dataset

```
Skill Recommendation System: Ratnesh Shah - 1401110
Please enter your skills: graphic design,autocad,animation
1st preference -->You are recommended to pursue design engineer profession
You are recommend to acquire following skills-->
-----------------------------------------------------------------
2nd preference -->You are recommended to pursue software engineer profession
You are recommended to acquire following skills-->
    'java'

    'python'

    'c'

    'sql'
```

Fig. 9: Result generated

VI. Conclusion

The methods have been tested using a set of public profiles extracted from the data provided, naturally subject to noise and inconsistencies; I only applied a couple of trivial preprocessing steps to them. I have tested both the modules with a smaller set of data and the results are promising for both.

VII. Future Work

My goal is to increase accuracy of recommendation, 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. Finally, we consider to test clustering and recommendation with more extended datasets, including more profiles and possibly further information for each, in order to improve the results for both tasks.

References

[1] A Hybrid Approach to Recommender Systems based on Matrix Factorization,Diploma Thesis at Department for Agent Technologies and Telecommunications Prof. Dr.-Ing. habil. Sahin Albayrak

[2] Job Recommendation From Semantic Similarity of LinkedIn Users' Skills, Conference Paper.January 2016, DOI: 10.5220/0005702302700277