Career Path Recommandation System

Algorithm and Optimization of Big Data (AOBD)

Winter 2017

School of Engineering and Applied Science

Ahmedabad University

Report by: Parth Satodiya (1401056)

***Abstract*—Recommendation system is used to suggest products or services that best fit with the user's personal preferences. I covered career path recommendation system in this paper. There are two modules, one module takes user's career profile and based on that suggest further skills to be consider. Second module take career goal from user and suggest career path. I train this system with some user profile data which is in Jason format. There are two algorithms proposed one for each module. At the end there are some outputs for each module.**

***Keywords*— dictionary, tokenize, lemmatize, json**

I. Introduction

In today's world, people mostly students are always facing the problem of choosing career path which suitable for his/her personality, achievements and environment. Building a best recommendation system is currently an ongoing research problem. We need to consider all peoples' personal preferences, skills and choices to build a system. We can also use some highly advance technology like machine learning, deep neural network, artificial neural network, natural language processing etc. In our problem, there are two possible break down.

(1) Predicting career,

(2) Predicting steps to get there.

In first problem, there are a lot of issues with the data as the skills and degrees have multiple variations and there are finite yet a lot of titles, similar issue with job titles etc as there isn't a standardized vocabulary for these things. If we have to predict a career, this essentially becomes a multi-class prediction problem because you have to determine out of 1000+ possibilities which career is suited the best. Now if you have to determine how good the career fits someone, it just can't be a binary classification, you need values along with class. This doesn't really leave you with a lot of choices. You can either train multiple traditional ML models to get results OR build a Neural Network to give you multiple predictions along with values.

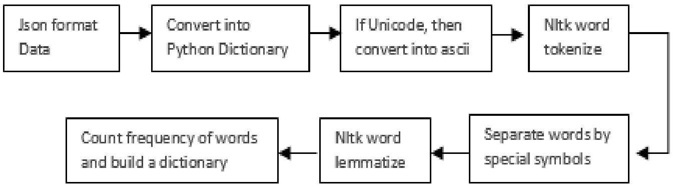
In second problem, we don't have the step that you have already taken. If user interested in careers which he/she could potentially achieve after taking certain steps, then we would need our model to predict a career 2 steps in advance and enlist you the steps. Other work around for this would be to take predicted careers with low scores and get statistical data on how to improve the score or better chances of successfully getting a certain role.

II. DATASET AND ALGORITHM

Following are the algorithms for both modules. I have used python language to implement these algorithms.

A. *Cleaning data*

For cleaning the data, I have used natural language tool kit (nltk) package in python. Following is the flow chart for cleaning data.

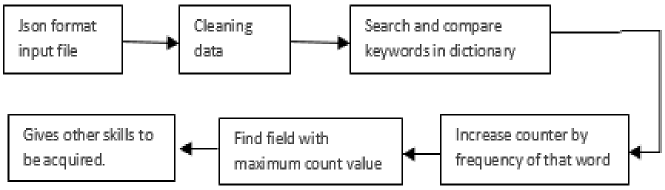

Json format Data
Convert into Python Dictionary
If Unicode, then convert into ascii
Nltk word tokenize
Separate words by special symbols
Nltk word lemmatize
Count frequency of words and build a dictionary

Our dataset is in .txt file with json format. So, first I convert these file to python dictionary using simple json package. Many of data are in Unicode format. So, I check for the Unicode data and convert it in equivalent ascii format. Then I used word tokenize function from nltk python package. It will separate all the words from the sentences and store into a list. Then I clean up the list and remove all special symbols.

Then I applied lemmatize nltk function. Lemmatize is the process of grouping together the different forms of a word so they can be analyzed as a single item. Then I have converted all the words into lower case and find the frequencies and store it in a python dictionary. I have repeated this process for all the dataset files and build a final dictionary which has separate dictionary for separate career option.

B. *Module 1*

In module 1, we take a user's profile as an input, analyze it and give the further skills that needs to be acquire in that field. Following is the flow chart for module 1.


Json format input file
Cleaning data
Search and compare keywords in dictionary
Increase counter by frequency of that word
Find field with maximum count value
Gives other skills to be acquired.

I first take a user profile in json format. Clean this data and store keywords in list. Then search all the keywords one by one in the previously build dictionary and if match then increase counter by the frequency of that key world in that particular career field.

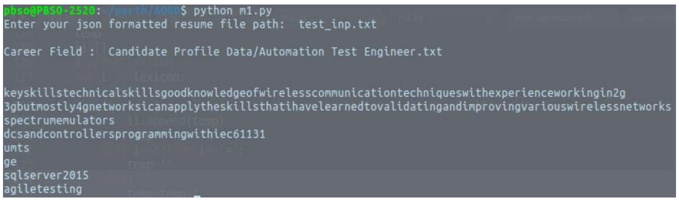
After searching, find the career field that has maximum counter. Find other keywords from the dictionary of that career field that is not there in the input keyword list and give it as an output of the first module.

C. *Module 2*

In the second module, we take career field as an input that the user wants to choose. Select the dictionary of that field and output the key worlds that has maximum frequency in that dictionary.

III. Output

Following is the module one output. I created one file named ‘test.txt; and input it in program. Program find the best match career field and gives additional skills that the user need to acquire.

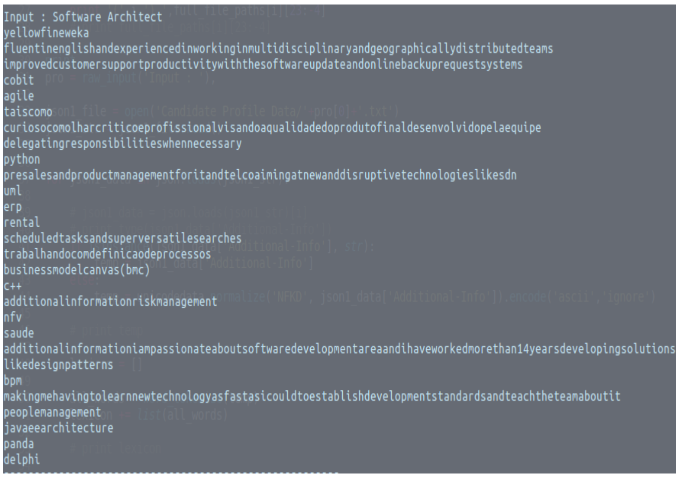


```
pbso@PBSO-2520:~/parth/AOBD$ python m1.py
Enter your json formatted resume file path:  test_inp.txt

Career Field :  Candidate Profile Data/Automation Test Engineer.txt

keyskillstechnicalskillsgoodknowledgeofwirelesscommunicationtechniqueswithexperienceworkingin2g
3gbutmostly4gnetworksicanapplytheskillsthatihavelearnedtovalidatingandimprovingvariouswirelessnetworks
spectrumemulators
dcsandcontrollersprogrammingwithiec61131
umts
ge
sqlserver2015
agiletesting
```

Below is the second module's output. I entered the career field and program gives some skills as an output that I have to acquire for that career field.



```
Input : Software Architect
yellowfineweka
fluentinenglishandexperiencedinworkinginmultidisciplinaryandgeographicallydistributedteams
improvedcustomersupportproductivitywiththesoftwareupdateandonlinebackuprequestsystems
cobit
agile
taiscomo
curiosocomolharcriticoeprofissionalvisandoaqualidadedoprodutofinaldesenvolvidopelaequipe
delegatingresponsibilitieswhennecessary
python
presalesandproductmanagementforitandtelcoaimingatnewanddisruptivetechnologieslikesdn
uml
erp
rental
scheduledtasksandsuperversatilesearches
trabalhandocomdefinicaodeprocessos
businessmodelcanvas(bmc)
c++
additionalinformationriskmanagement
nfv
saude
additionalinformationiampassionateaboutsoftwaredevelopmentareaandihaveworkedmorethan14yearsdevelopingsolutions
likedesignpatterns
bpm
makingmehavingtolearnnewtechnologyasfastasicouldtoestablishdevelopmentstandardsandteachtheteamaboutit
peoplemanagement
javaeearchitecture
panda
delphi
```

References

[1] https://www.analyticsvidhya.com/blog/2016/06/quick-guide-build-recommendation-engine-python/

[2] https://www.quora.com/What-would-a-career-recommendation-engine-look-like

[3] http://www.gdeepak.com/thesisme/Applying%20Data%20Mining%20For%20Job%20Recommendations.pdf

[4] https://www.researchgate.net/publication/269272063_Career_path_recommendation_system_for_UiTM_Perlis_students_using_fuzzy_logic.

[5] http://stackoverflow.com/questions/30685363/python-csv-to-dictionary-columnwise.

[6] http://www.cse.unsw.edu.au/~bradfordh/papers/pricai14.pdf.

[7] https://pythonprogramming.net/preprocessing-tensorflow-deep-learning-tutorial/.

[8] M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.