Florida International University
Knight Foundation School of Computing and Information Sciences

Introduction To Machine Learning
CAP 4612

Final Report

Project Title:
Predicting tech jobs demand based on the skills

Team Members: Fares Amamou

Instructor: Dr. Mustafa Ocal

Table of Contents

INTRODUCTION:

Motivation:

In the past few years, the available jobs related to tech have been changing dramatically and in fast pattern specially after the recent layoff in tech company. As highly demanded position become unavailable in just few years and replaced by others. For students and IT task force it can be hard to keep up with the requirement. That is why it is important to know what is going to be necessary in the future. Therefore, the Tech Jobs Prediction model build for this project is crucial to keep up with those changes. It will predict the available jobs in the future and their salaries based on the skills and roles needed by the market.

Related work:

In the research paper “Job Prediction: From Deep Neural Network Models to Applications”, kietnv, ngannlt, and anhngt used the dataset for IT job prediction proposed by Papachristou. It is a dataset that consists of 10000 distinct job descriptions collected from the online finding-job sites, annotated with 25 different types of IT-related job. The attribute of that data is the job description and job title which is nominal, and it was pre-processed by converting the job description into lowercase letters, deleting special characters, transform the job description into a set of words, removing the stop word in the descriptions, and then representing those words into vectors with pre-trained word embedding sets. However, no data normalization was used.
Now when it comes to the implementation, three deep neutral network models were used. First, the TextCNN model which is a deep learning algorithm that proved its efficiency when it comes to Natural Language Processing, and it was used for emotion recognition, sentiment analysis, and question classification. Second, the Bi-GRU-CNN model because it was already used to predict wages, based on job descriptions such as job content, job requirements, working time, job

position, and type of job. So, it is related to the problem being addressed in this paper. Finally, the Bi-GRU-LSTM-CNN model as it was ranked the fifth on the scoreboard on the public test. Eventually, the model created achieved a high F1-score of 72.71%. However, some difficulties were faced due to the fact that the dataset contains very limited data with 10,000 annotated job descriptions.

Citation:

Tin Van Huynh, Kiet Van Nguyen, Ngan Luu-Thuy Nguyen, and Anh Gia-Tuan Nguyen. (Dec 27, 2019). Job Prediction: From Deep Neural Network Models to Applications. University of Information Technology, Ho Chi Minh City, Vietnam 2Vietnam National University, Ho Chi Minh City, Vietnam. Retrieved from https://arxiv.org/pdf/1912.12214v2.pdf.

DATA:

The following section provides the dataset used descripted in details along with the link to access it.

Dataset:

The dataset used for this model is U.S. Technology Jobs on Dice.com collected from Kaggle datasets. It consists of 22000 unique objects of attributes: The page URL, the name of the company, The type of employment, The description of the job, The ID of the job given by the company, The address of the job, The title of the job, The date of posting, The timing or the shift of the role, and the name of the site. The target value will be the job title and the category of demand of that job.

Link of dataset:

https://www.kaggle.com/datasets/PromptCloudHQ/us-technology-jobs-on-dicecom

PLANNING AND ADJUSTMENT:

This section describes the planning and what actually went into the realization of this project. This section also describes the reason behind the change in plans.

Planned work:

When it comes to data preprocessing, the first thing is dimension reduction to remove most of the attributes and leaves only the job description, location, and job title. Then, data cleaning to complete missing values, fix incorrect data, remove inconsistent data. And no data normalization is needed.
The machine learning model that is going to be implemented is the K-Nearest Neighbors (KNN) algorithm using C++ standard library.
For the system evaluation 70% of the data is going to be used for training and 30% for testing. And accuracy, precision, recall, and F1 score are going to be referred to determine the success of the model. To avoid overfitting, I will also perform k-fold cross-validation.

Implemented work:

Now what was actually implemented went quite different. First, for data preprocessing, the dimension reduction was still used to the remove the majority of attributes. But it ended leaving the job description, location, job title, and the skills. Also, data normalization was added to better visualize the data.
The machine learning model implemented was K-means clustering algorithm after performing the Elbow method to determine the optimal number of clusters K using Python.
For the system evaluation since it is an unsupervised learning model, the only choice left was human validation.

Reason behind the change of plans:

At the beginning, I was thinking about labeling the dataset myself by computing the number of repetitions of some job title in the data frame and clustering all the object into four categories based on the title present among the data: Low Demand, Average Demand, High Demand, Very High Demand. Then, use it to train the model to be able to categories the rest by itself.

However, after going over the dataset, I realized that plenty of jobs have different titles but with the same job description. Also, some jobs have the same titles with different job description. So, I concluded that the title can not accurately determine the job demand. What actually mattered was the skills required for that job.
So, I decided to build an unsupervised learning model that can label the dataset by itself based on the skills of each job.
Also, Python was used instead of C++ due to the presence of wide Machine Learning libraries that can help build the model.

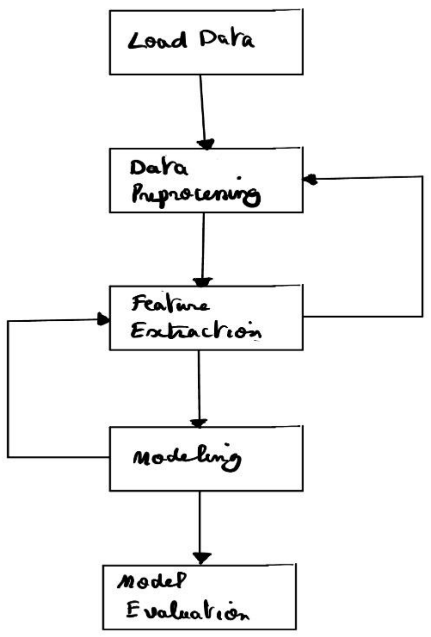
IMPLEMENTATION:

This section describes the implementation section related to the libraries used, pipeline, data preprocessing, feature extraction, classifiers, and performance measurement techniques.

Used libraries:

1-NumPy library
2-scikit-learn library
3-Pandas library
4-Matplotlib library

Pipeline:



Preprocessing:

The data preprocessing stated with dropping most of the columns and leaving only job description, location, skills, and title. Followed with min-max scalar as normalization to better visualize our data.

Features extraction:

For feature extraction, I decided to transform the skills into numerical features that can be processed while I preserved the information in the original data set. It was done by searching for the most common tech skills which are SQL, Java, Python, Linux, JavaScript, AWS, C++, C, C#, .NET, Oracle, HTML, Scrum, Git, CSS, ML, Azure, Unix, SQL server, Docker. The skills category was converted into a string array and traversed using a for loop computing the number of repetitions of each skill. Then, that number was assigned as the skill weight because the skills that were present most should have higher weight since they are the most required ones. After that, the skills array was traversed again using a for loop to compute the score of each job (the sum of the weights of skills presents in that job) along with the number of skills present in each job and store them into two integer arrays. Then those arrays were added to the dataset to be processed later on.

Classifier:

As for the classifier implementation K-means clustering algorithm was used with K equal to two after determining it from the Elbow method. It classified the objects based on the number of highly demanded skills presents in their skills attribute along with the importance of those skills.

Performance measurement:

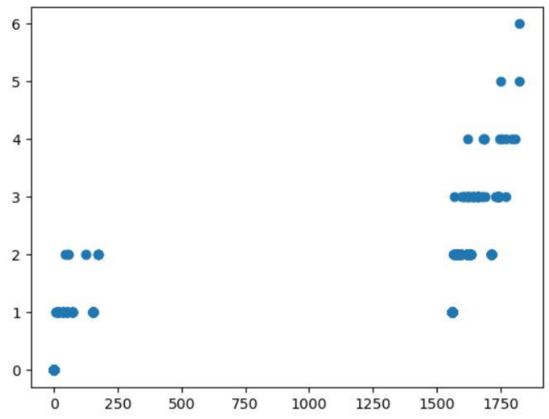
Now for performance measurement, since this is an unsupervised learning model, a human validation is required as it is the only way to determine the efficiency of the classifier.

RESULTS:

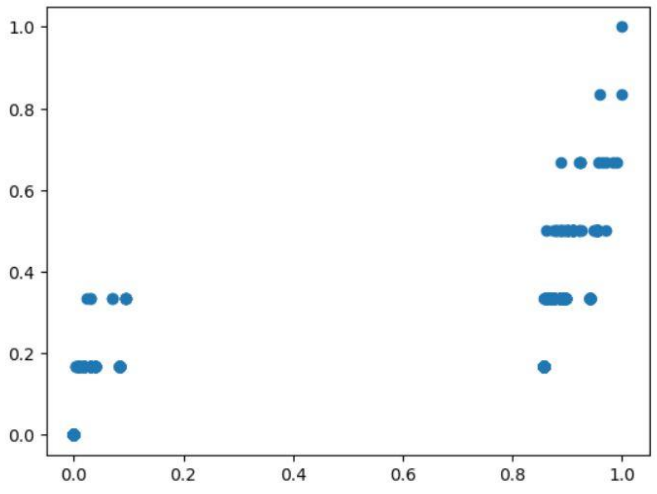
This section shows the plots obtained from data visualization and compare it to the previous system.

Data visualization:

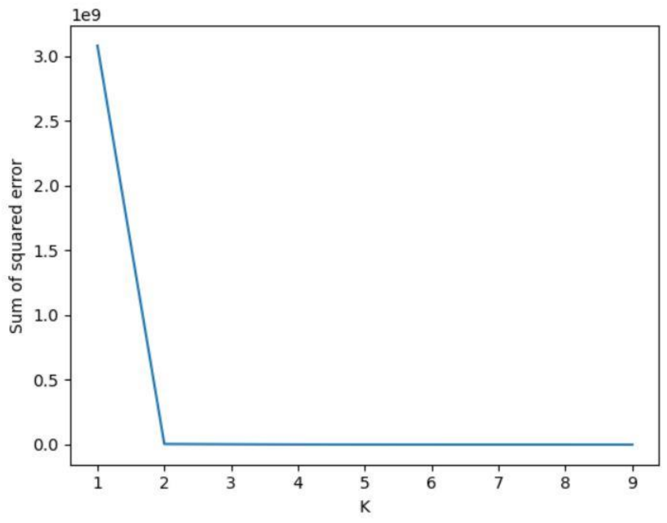
1- Scattered plot of skills score vs number of skills:



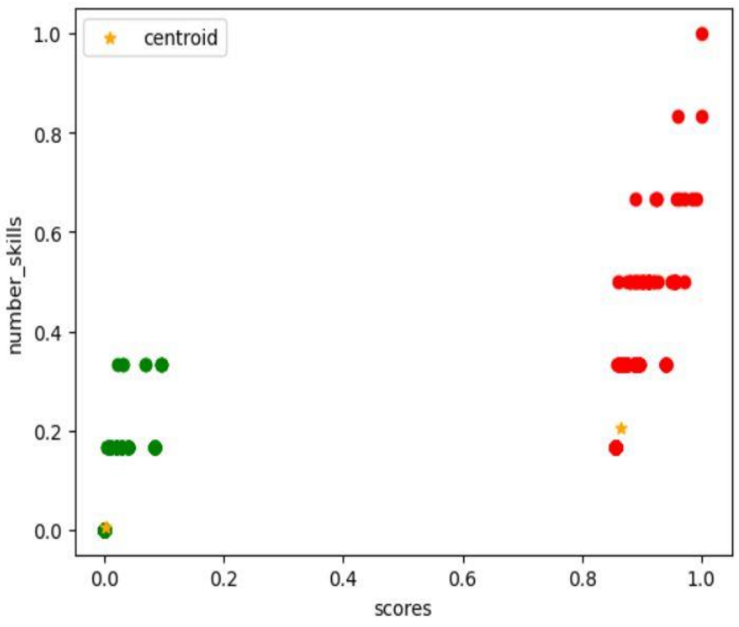
2- Scattered plot of skills score vs number of skills after min-max scatter:



3- Scattered plot of k vs sum of squared errors:



4- Scattered plot of clustered data:



Difference from the previous system:

The previous system used job description as input and output the job title. Even though, this can be helpful determining job. It has nothing to do with determining the demand of a certain job. Unlike the current system that takes a dataset of jobs as input and classify it into high demand and low demand.

CONCLUSION:

This section describes how the system performance went, the reason behind it, future work to improve the project, and my contribution to this project.

System performance and reason behind it:

Even though evaluating the system performance was limited due to the absence of system validation methods for unsupervised learning, the system did a decent job classifying the jobs into two categories. However, they were some limitations and unconsidered issues that can affect the efficiency and precision of the system. For instance, the scores and total number of skills of each job was based only on 20 specific job and rest were ignored. So, they may be some other important skills that were neglected. Also, some jobs had a NULL skill attribute which affects the results too.

Future work:

To address the issues mentioned and improve the model, some future work is needed. I would suggest instead of just looking for the number of present of only those 20 skills, the system can look for any skill that was present in the dataset. Then compute the number of presences of those skills found. Also, after computing the score and number of skills of each job, the model can replace the score and number of skills of the jobs that have NULL for skill by the mean of the rest of the jobs.

Contribution:

This project was made entirely by Fares Amamou without any contribution from any other person or source.