ANLY-533-01

Fall Term – Group 3

Adam Sampson, Jack Letcher, Nathan Thomas, Taylor Maxson

## Business Purpose

The business purpose defined for this project is to develop a method, using text classification, to produce a list of the top fitting jobs for a client of a career recruiting firm, with the intended goal of getting the client selected for an interview. Initially, this concept would review a client’s resume, scan for the key words and pair it with jobs matching the highest number of key terms. This model provides a solution to the problem of spending several hours of work time sifting through job postings and determining which jobs accurately match the skills of the client.

During the brainstorming sessions, several secondary applications to this model were developed and discussed. One such method would include reverse matching, or determining which key words or skills should be attained or highlighted on a resume in order to more closely match the skills needed in the job posting. Another potential use xxxxx

## Data

The data used for this project was scraped from the individual companies’ websites. The original strategy for obtaining data was to do an API call to job sites such as Monster.com and Indeed.com. These sites though did not grant permission to use the data, therefore the group decided to scrape individual job data from company websites. The parameters chosen were to perform a search on the “careers” section of the company website using only keyword “Data”. The simplicity here would provide consistency between each company. The extracted raw files were in XML and HTML formats and were formatted differently for each company. The five companies who we were able to obtain job data for were Apple, Dice, Humana, Microsoft, and Walmart.

## Data Preparation

Obtaining the data was a tedious process as each website was built a bit differently and code had to be tweaked to scrape the different sites. The data sets themselves presented some real challenges as well. The first method to getting it to a useable corpus was to parse the xml and html data into a data frame. It was determined to keep only a few pieces of identifying information from the documents and load most of the remaining data as the description. The information kept was the job identification number, the job title, and finally, the job description. The job description consisted of text about the job, the requirements for the job, as well as any company information that was included in the file. A combination of R functions and regular expressions were used to extract the information from the documents. There were a few issues with the Dice data though. The description data was duplicated on several sub-nodes, causing confusion on what sub-node to include. The simple solution was to pull in sub-nodes 18-20, however some of the descriptions ended at node 18. The data was able to be extracted to a text file and the data string from the text file was read into the description field. One other key issue with the Dice data was that the job title was not able to be parsed from the data. On the other data sets, the job title was embedded as a separate node, while in the Dice data, it was embedded in the general text and not able to be parsed out. The data was eventually cleansed to include only alpha text, numeric text as well as the “+” and “-“ symbols. The final transformation was done in the model files, where the text was transformed to lower case, stop words removed and whitespace stripped out. The final data was then loaded into a corpus used for the modeling process.

## Modeling

In an attempt to produce a model that would accurately match a resume with

### k-Means Clustering

A screenshot of a cell phone

Description generated with high confidenceOne method chosen to analyze the data was k-Means Clustering. This method was chosen to group jobs by types so that a client could match key terms from their resume to the terms in the job descriptions. Initially, a histogram of the top words was generated to see the highest key term frequency. Intuitively, data was the highest frequency term since our job search was based exclusively on this term. Other high frequency terms were not helpful in determining groups associated with “data” job types. There were some key qualities or traits that were more frequent, such as “management”, “engineering”, “technical”, and “software”.

A screenshot of a cell phone

Description generated with very high confidence

A simple word cloud was produced to graphically display the frequency of words in the corpus. The larger words indicate the highest frequency.

The expectation of this model was to gain insight into what categories may surface from the key terms.

A screenshot of a cell phone

Description generated with very high confidence

### TDF-IDF

### Topic Modeling (LSA)

### Jaccard Similarity

## Evaluation

Did we solve the business problem?

Further analysis could be done on the frequent terms to determine what other key terms could be used to scrape jobs outside of just the word “data”. This would allow us to capture jobs that are similar, using some of the same methods listed in our report, which may not contain “data” as a key term.

## Deployment