Client Mentor: Mr. Surendra Dattatrey

Mentor: Prof. S M Bendre

Project: E-recruitment using ML

Core Team:

Vikash S Negi - 11810048,

Devender Bansal - 11810089,

Atul Sharma - 11810126,

Sandeep Kumar Singh - 11810014

**Table of Content**

1. Executive Summary
2. Motivation
3. Project Description
4. Process
5. Model Comparison
6. Conclusion/Recommendations

**Executive Summary**

Hiring right talent within short time is one of the biggest challenged being faced by the organizations. Failure to do so may hamper the business growth as well as competitiveness of an organization. One of the biggest challenges in achieving it is Manual Shortlisting of high number resume applications.

The manual shortlisting needs long time, large efforts & is very expensive. This project is an attempt to explore and provide a way to shortlist the high number of resumes automatically using modern machine learning & Text Analytic tools resulting in shorter hiring cycle and cost reduction for an organization.

The Expected outcome of our Project is a Machine Learning model which process each received resume and classify it as Accepted or Rejected along with the summary of its features.

**Our Approach for the Project is as below**

Exploration of what features are important for evaluating a resume

* Through discussion with HR recruiter
* Through EDA on set of resumes provided by Client

Automated extraction of those features

* Through employing ML & Text Analytic Techniques on the provided resumes

Building Supervised training model

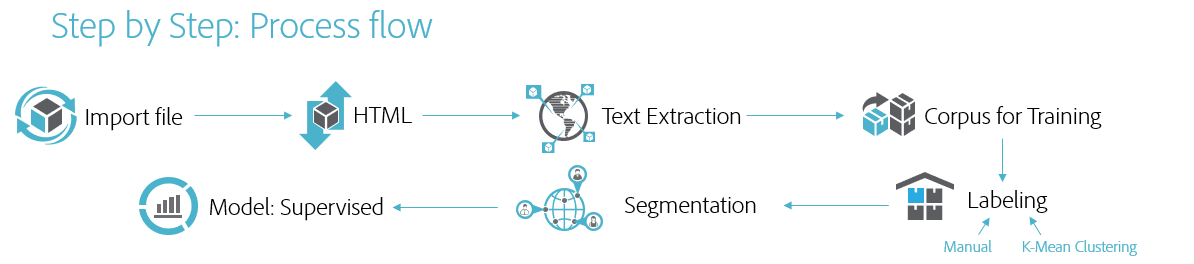
* Basis Labeled resumes

Classifying the resumes basis, the supervised model

Our Goal for building the model is

* Build a Generic model so that it can be applied to any Job posting
* Build a modular framework, so that individual models can be improved upon, continuously

To achieve the Expected Outcome, the approach & Goal, we have created a framework for this model and followed a stepwise process low to achieve the desired outcome.



**Motivation**

**Why we are doing this?**

One of the biggest challenges facing HR professionals today is finding the best talent to hire. This task has proven burdensome in the past, in part due to inefficient manual tasks that plague the recruiting process and the lack of access to the right data to make informed decisions. In fact, per research conducted by LinkedIn, 46% of recruiters and hiring managers have identified "finding the right candidate" as the biggest hurdle in hiring today. To tackle these challenges, new technology companies are rapidly emerging in the HR tech ecosystem with robust solutions that use Big Data, predictive analytics, and AI to automate and improve everything in the recruitment process from job advertising and resume screening to applicant engagement, scheduling, and recruiting by text. These new tools offer us ways to help overcome the limitations and biases inherent in recruiting with automated processes that are hyper-responsive to market data, complex metrics, and even budget constraints. Hence this project is to find out on very first problem, where we would like to find out a solution to match job requisition with candidate’s resume.

**Project Implications from industry perspective?**

As we know, screening the resume is most time taking exercise in recruitment process across industry as it takes up to 23 hours for one hire approximately. In a real-life example when for a job opening receives 400 profiles (Avg basis) out of which 80% to 90% out of them are not eligible for the job, one can imagine why most talent acquisition leaders are still find hard to recruit the right candidate from a large applicant pool. Through this project we will try to cater an industry wide problem using robust automated system to provide the top candidate for a job from the applicant pool.

**Why this project is important to sponsor?**

In current scenario, high number of fresh graduates, huge application volume for a single job posting is all time high which directly impacting HR KPI’s. The entire hiring process is slowed down due to these problems impacting not only on revenue but also on the quality of hire candidate. HR needs to finalize few resumes through manual screening process from the large pool of applications. This project is directly impacting the recruitment process of the client organization along with huge time and cost benefits impacting both top and bottom line of the organization.

**Project Description**

**Business Context**

Bringing the right people on board is a major part of long-term business success. Employees are any company’s most important asset. However, finding right talent through existing manual and keyword matching techniques are a big pain for recruiters. It involves too much effort, waste of valuable which in turn leads to in-optimal hiring.

As the outcome of the project, we want to automate the process of shortlisting of resumes.

That entails shortlisting using a pre-trained model which go beyond the simple technique of Keyword Matching.

Some of the tools used relies heavily on Keyword Matching, which while address the automation challenge. But at the same time are unable to extract the relative strength of a given feature.

In our Project, we intend to use advance text Analytics techniques like NLP with various Machine learning model to extract not only the Keywords but relative strength of the required feature.

**Area of impact**

Recruitment across organization, HR department, Hiring Managers will be the area of Impact. Reduction in hiring cost, Time reduction in resume/profile screening process.

Along with the Internal HR & recruitment departments. This project can be further developed in providing commercial resume shortlisting as a service by various Job consulting firms as well as Online job portals

**Situation Analysis**

* **Current Situation:** The resumes/Profiles have been screened manually with respect to the job description which is really a cumbersome job.
* **Proposed Approach:** A machine learning recommender program will be built to provide top profiles based on the job description passed to the program
* **Desired Situation:** An Automated engine which goes beyond keyword search and help match the right set of resumes for a given Job description.

**Background Study Required**

We divided our background study into two parts: Domain Knowledge & Execution

**Domain Knowledge** is the key aspect of this project. We spend number of hours discussing with our HR partner on how the recruitment process works, what are the different ATS tools are available, how the current screening process is executed, what is the average time to complete one job posting, and many other relevant information is captured.

**Reading different types of formats:** This become a challenge since resume never follow a similar structure and is different from each other, while sharing common few heading/segments it can be vary from text to image. We put our sincere efforts read through and find patterns for input format. We excluded info-graphic & image-based resume for this project.

**Resume Analyzing Process:** Selecting parsing methodology, for segments extraction is required a lot of pre-reads on text analytics. Going through concepts on NLP approach, Machine learning approach was required.

**EDA with available set of resumes:** We undertook the EDA process with the set of resumes shared with us. Using Various text extraction and text analytic techniques, following questions were answered

* What are the most common headings used in Resume?
* What percentage of resumes has minimal one Table structure?
* What are the Unique words which are used in specific section of a resume?
* What are the common ways of mentioning total experience in Resumes?
* What are the unique words used in Education section of a Resume?

The above Background Study helped us to identify important Features to be extracted and we finalized 5 Features to be extracted is as follow

* Total Experience
* Relevant Experience
* Skill Level
* Education level
* College Tier Level

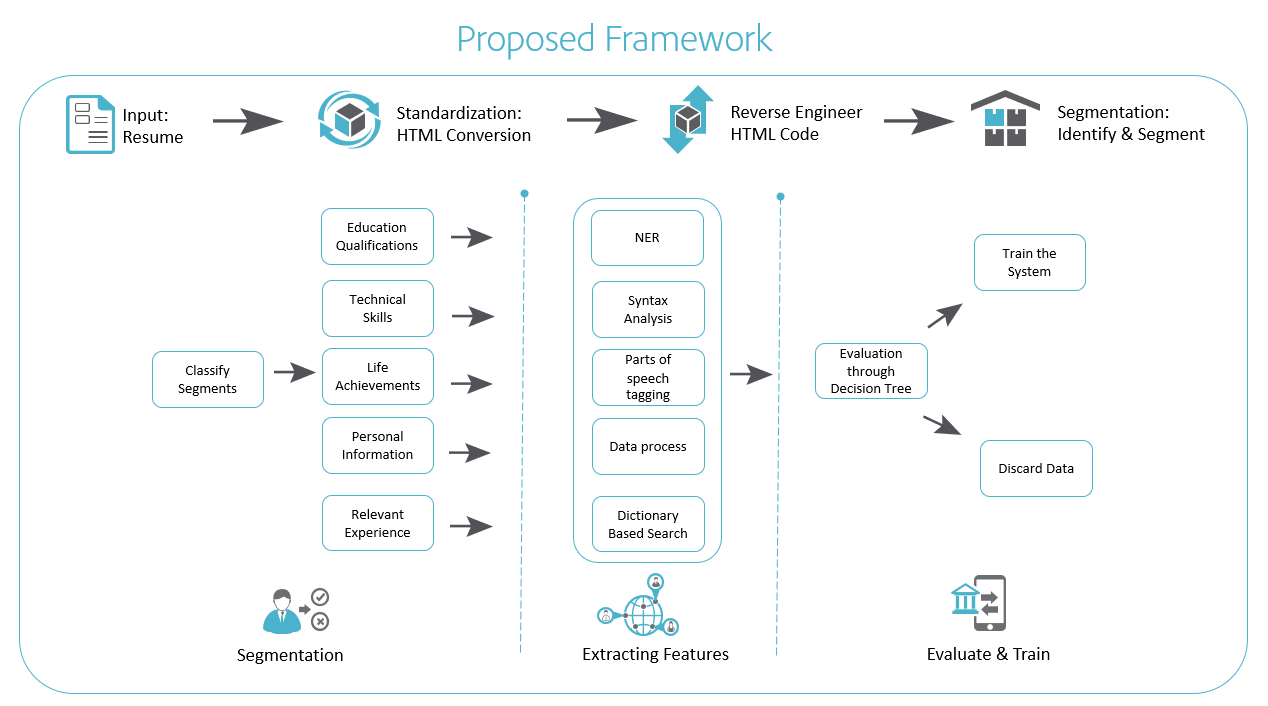
**Data Availability**

* Data is provided by client, as requested. We have received the data in form of resumes/profiles well within the first 4 weeks.
* The data is limited, as we only received 80 resumes to work on this project from the client along with 4 JD’s to work with. More profiles might help us in training the model
* The resumes were masked for the PII information. Hence, we did not work on creating any feature pertaining the personal information.

**Limitations**

* Due to GDPR, resume was masked, limiting on the sharing the information
* Due to time constraints we limited our scope and only word doc and Pdf as an input format is considered for this project (exclude Image based resume).

Project Framework/Architecture



After acquiring domain knowledge, understanding current industry outlook, reading through various model, theories around job search, understanding resume formats this framework is proposed. This framework will be generic in nature and based on different machine learning algorithms, classifying techniques, text standardization, statistical models.

Keeping this framework as base, we break the entire process flow in step by step process. Since, each step has its own challenges while writing/executing codes we decided to follow divide and concur approach for our project.

Step by Step: Process flow

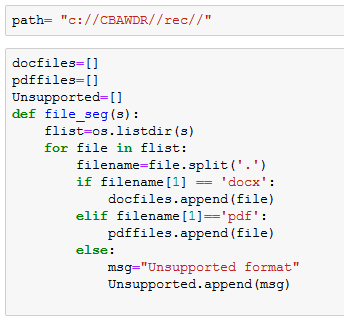
**Step 1:**

**Import file (Reading/Managing Input Data):**

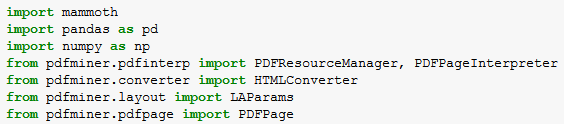
We decided to build our model on Word doc (Docx) and Acrobat (Pdf) format. As these are two widely used resume formats, we are excluding info-graphic and image-based resume.

We started with Segregating the set of .pdf & .docx resumes into two different lists. This was done so that we can use different libraries as needed for processing in subsequent steps

Technique & Codes (Sample Snippet Code)



Important Packages:

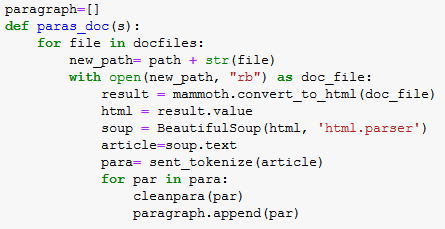


**Step 2:**

As a second step, we have converted the input resumes into HTML file. As during our EDA, we found out that many resumes had tables.

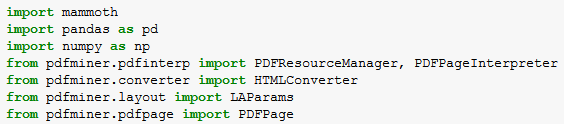
* Converting to HTML helped us to extract the table content without too much of complexity
* The HTML format also gave us the flexibility to use HTML tags while extracting text.
* We also could use an important library Beautiful Soup in Python for extracting text once format was converted to HTML

Technique & Codes



Important Packages

We have spent lot of men hours finding the best suited libraries for our requirement. We decided on **mammoth** for docx to HTML conversion & **pdfminer.six** for converting .pdf to HTML owing to simplicity and non- availability of single library to work on both

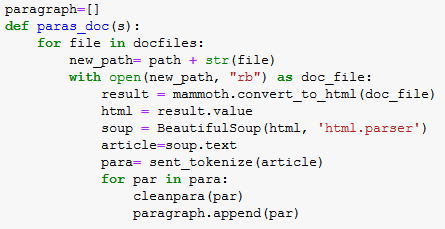


**Step 3:**

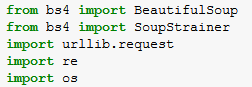
In this step, we started extracting the text from the converted HTML formats. The Text extraction was a regular process and it was undertaken for various purposes

* EDA process – We used The Text Extraction at EDA phase using the Beautiful Python library and using HTML tags
* Building corpus for labeling the paragraphs in the next phase
* Extracting Text from unlabeled Resume
* Extracting Text from Job Description
* Technique & Codes

The Soup. Text technique along with Sentence Tokenization was employed to get the output as paragraphs.



Important Packages



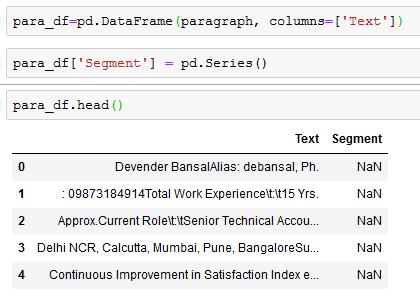
**Step 4:**



In this step, we created text corpus for training. We employed the aforementioned Text Extraction technique on a set of 48 Resumes. The output of the resumes was a list of 550+ paragraphs.

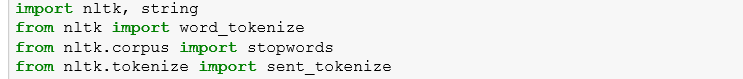
This is further be used for segmenting the unlabeled resume. Since, we faced challenge in segmenting the data into required segments through any method.

Sample Output

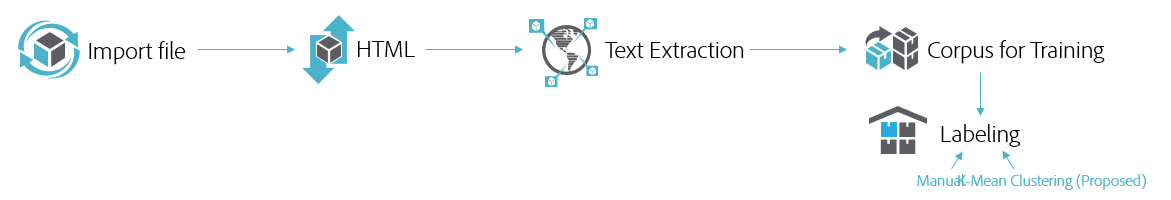


Important Packages

We used NLTK package for sentence tokenization while Pandas was used for creating Data Frames



**Step 5:**



In the Next step, we decided to Label the extracted paragraphs from 48 resumes.

This was needed to build a supervised model for paragraph Classification. A Supervised Paragraph model would help us to classify the text of a resume and JD into predefined segments.

To Label data, we had two options

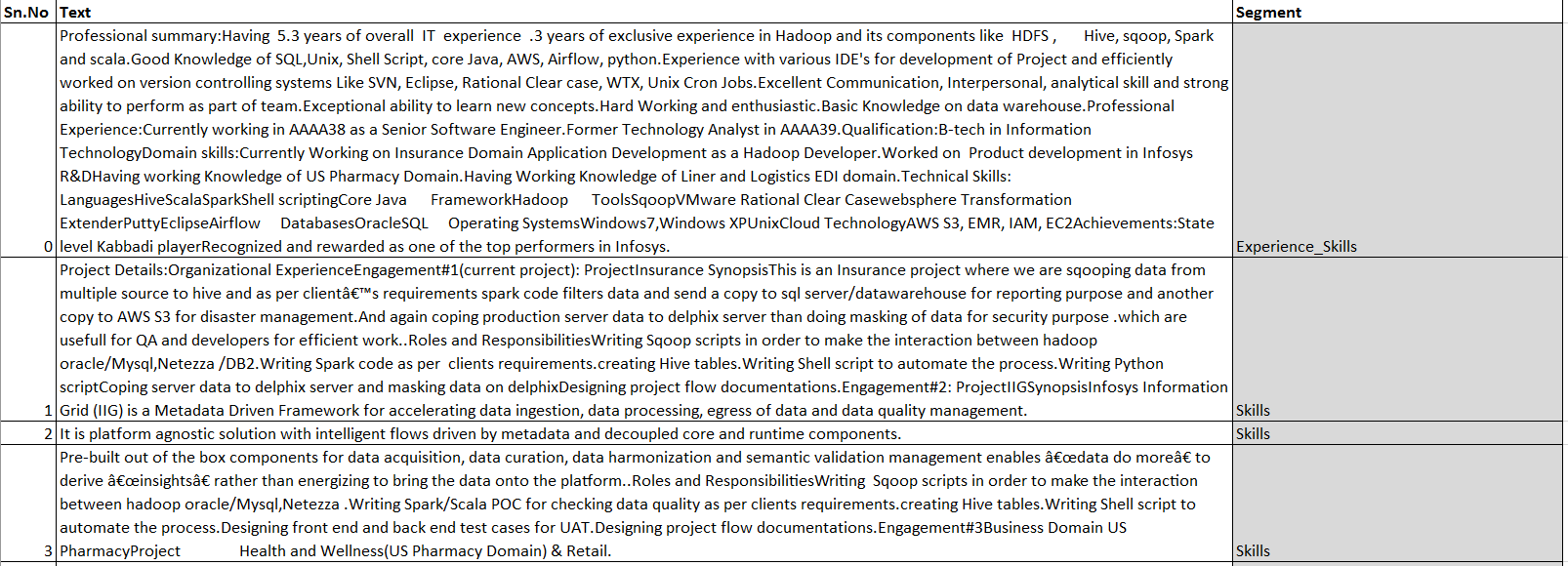
* Manual Labeling
* Employing Clustering Techniques like K-Means

Considering the low number of resumes and total of only 550+ paragraphs to be labeled. We went ahead did the manual labeling.

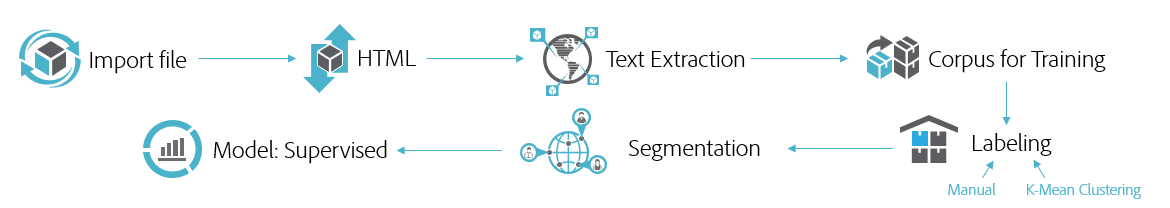
However, considering our project is open for improvements. It is strongly recommended to increase the number of resumes and employ Clustering technique to label the Paragraphs

Technique & Codes

* Manual Labeling
* K-Mean (Proposed)



**Step 6:**

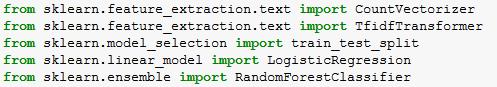


In this Step we built a Text Classifier from the training Data. We first converted the paragraphs into vectors. And then trained using Logistic Regression Classifier.

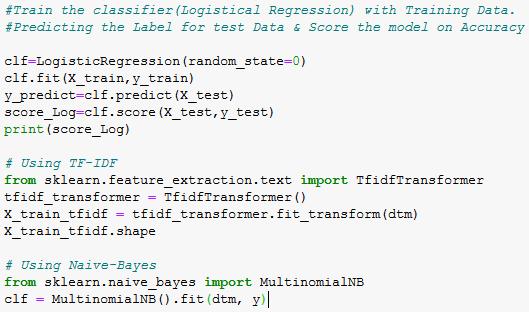
We test with other classifiers including Linear SVC and multinomial Naïve Bayes. However, as we received better accuracy on the validation set. We went ahead with Count Vectorizer and Logistical Classifier.

As this is the training model and we require the same vectorizer to be fit while prediction for new resume. This Vectorizer was saved for non- label paragraphs

Important Packages



Technique & Codes



Once our Paragraph classifier supervised model was built. It paved the way for us to Segment any resume or Job Description document into predefined following segments.

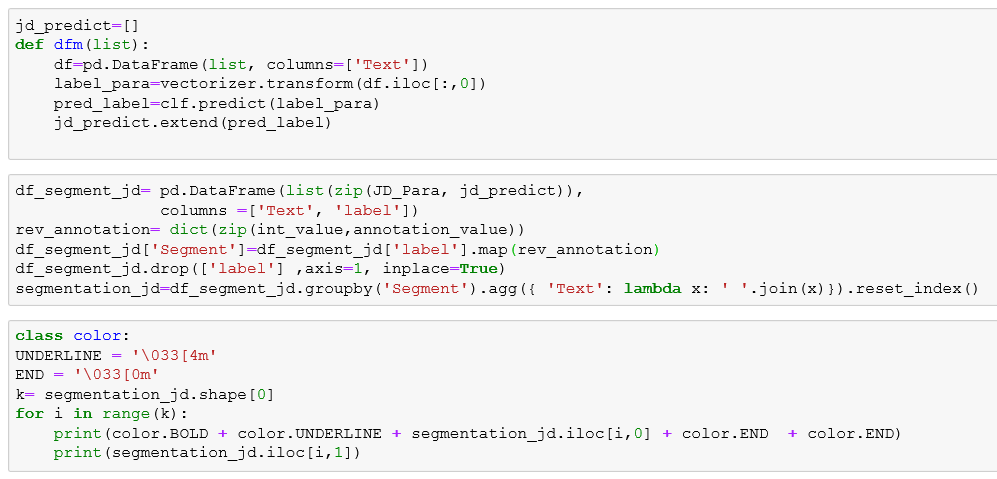
* Personal Details
* Education
* Skills
* Experience
* Achievements
* Others

Insert Supervised Model



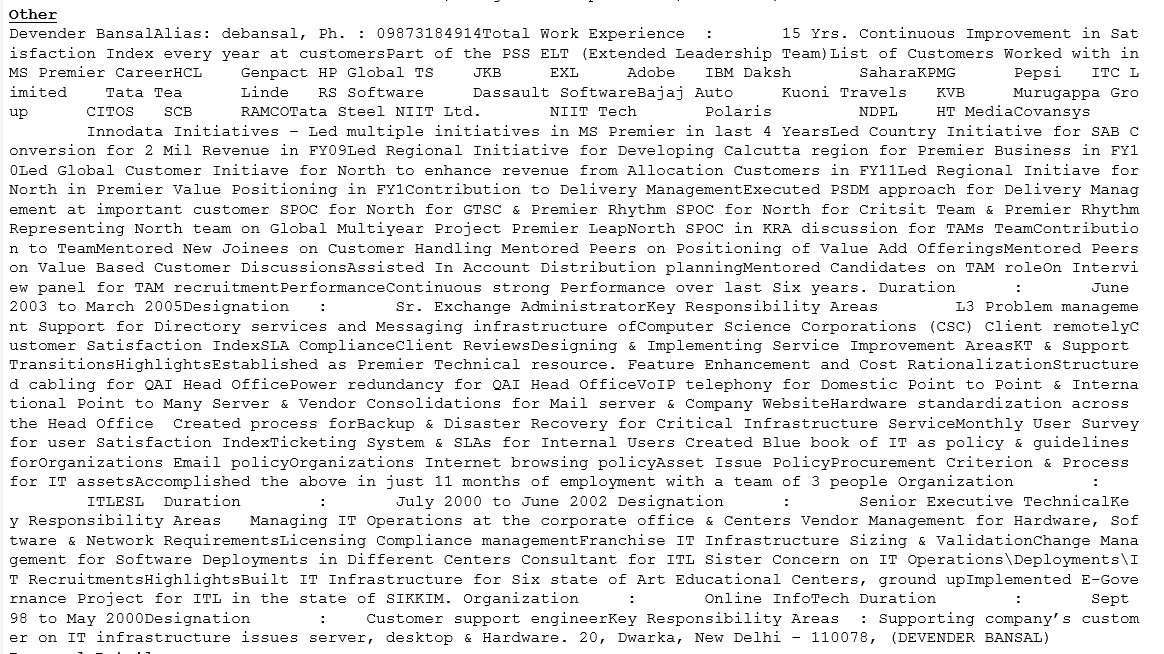
Insert Prediction Code





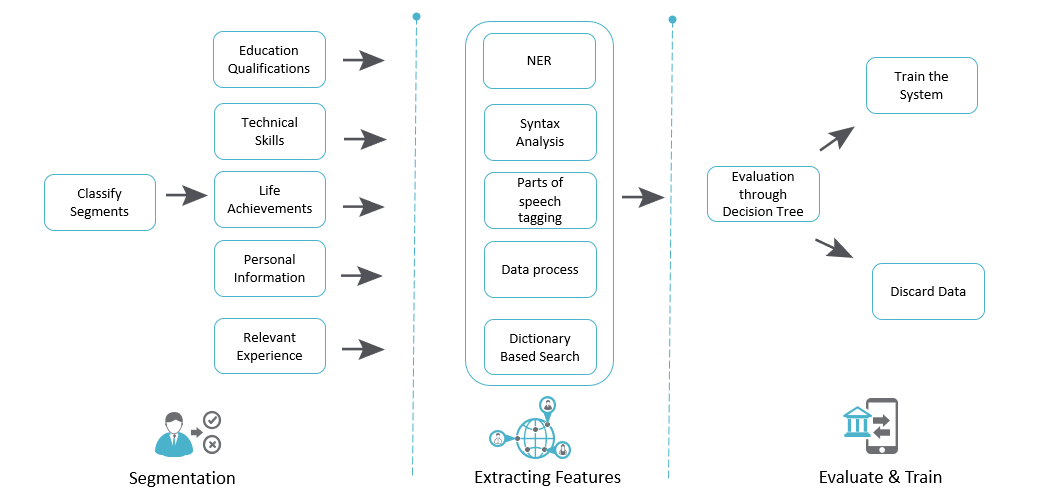
Insert Segmented Output







Post Segmentation Phases



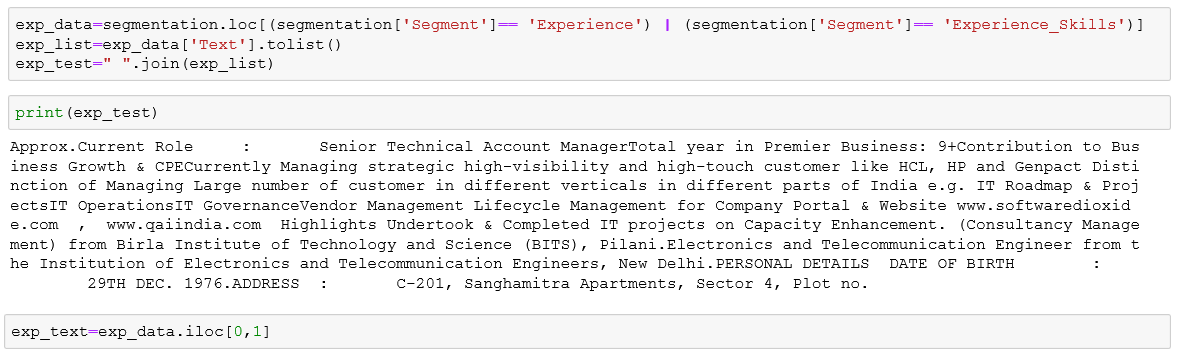
After Segmentation, we have output for a resume neatly converted into different paras for respective segments.

We then built below individual functions to be applied on different sector to extract the pre-decided five features

Experience

No of Years of Total Experience

We used the regex, post- tagging and Entity extraction to find the total or overall experience of a resume. The codes is as below



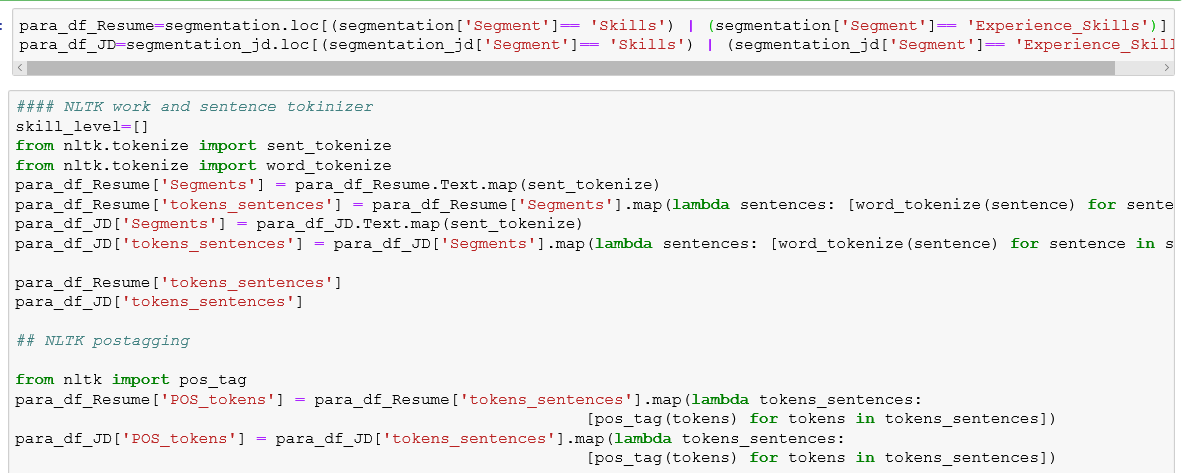


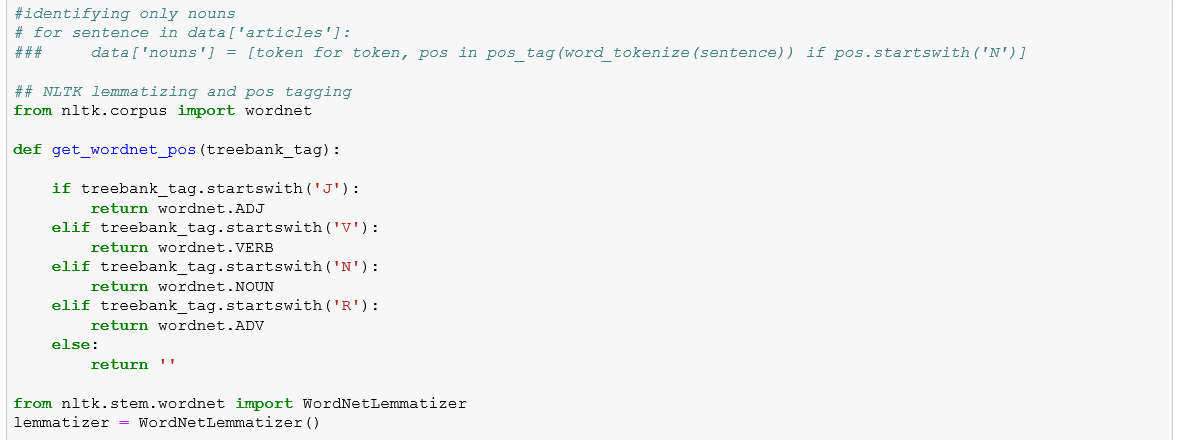
The Output of this function is a List with one value for each resume parsed. The output is numerical approximated year downwards.

Relevant Experience

We used unique Keyword matching techniques between the Skill section of Given JD with the Experience section of the Resume. It helps us to provide a matching score basis the similarity.

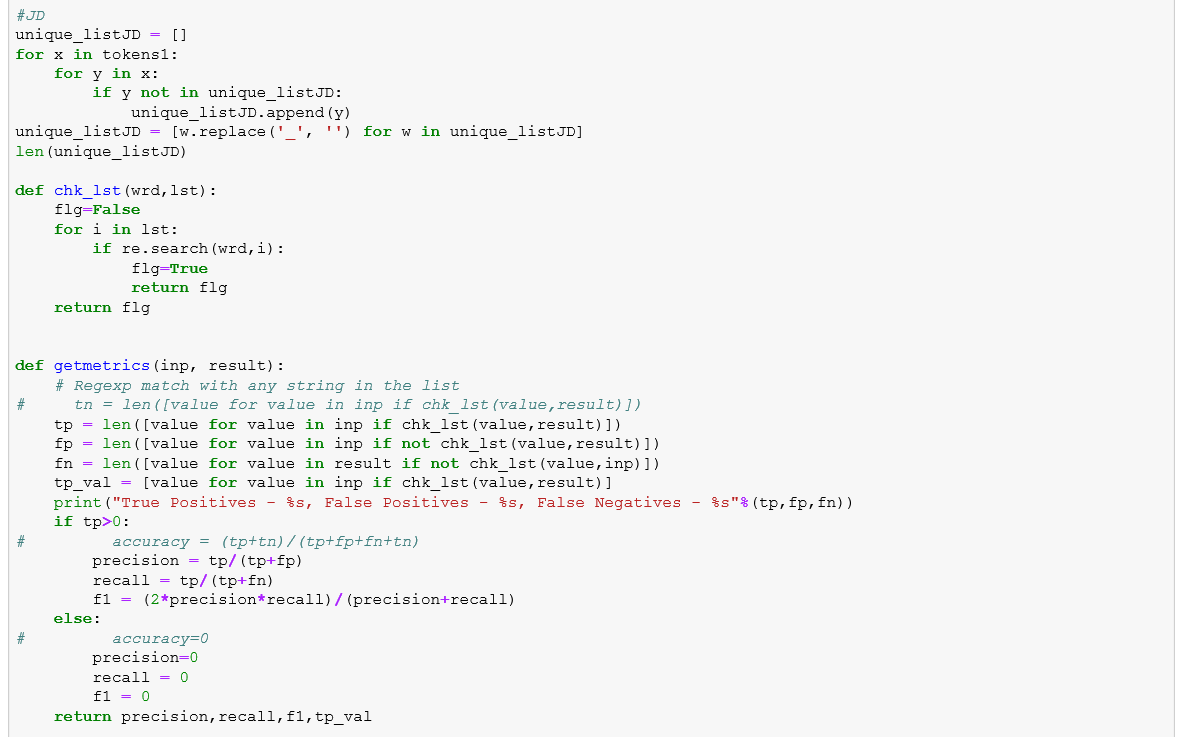
We employed Tokenization, Pos tagging, Lemmatization & similarity algorithms to understand the similarity score.

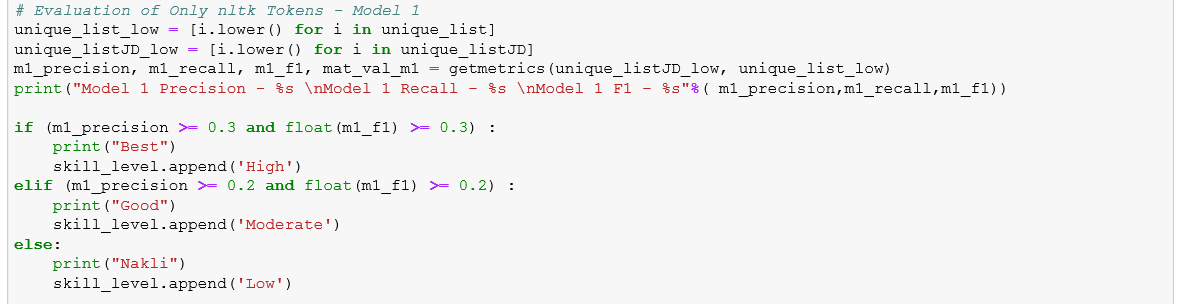










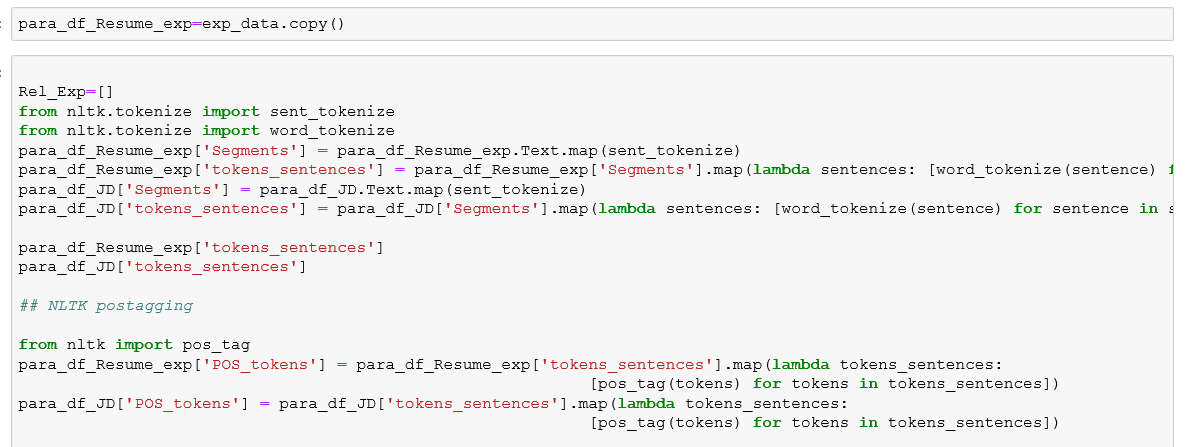


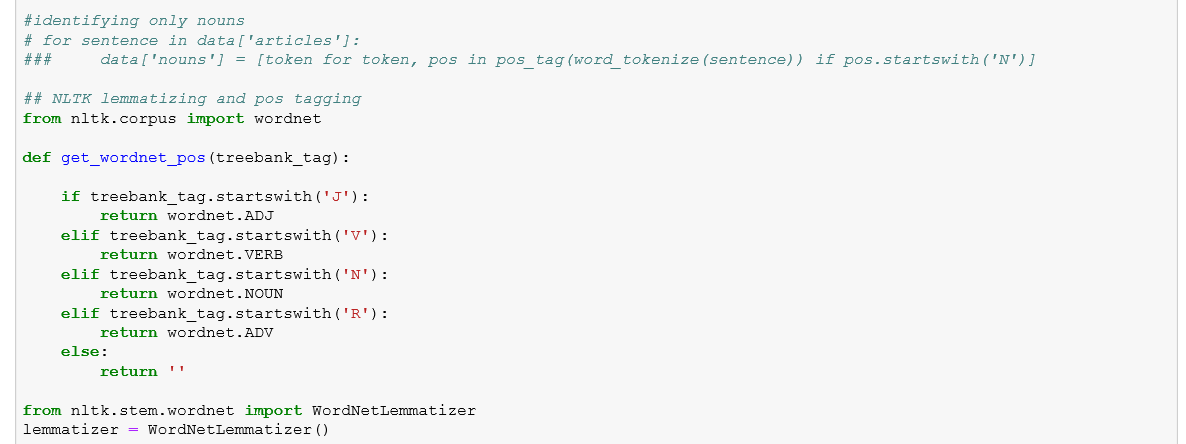
Output of this function is list of one relevance value for every parsed resume as either High, Moderate or Low

Skills Feature

We used unique Keyword matching techniques between the Skill section of Given JD with the Skill section of the Resume. It helps us to provide a matching score basis the similarity.

We employed Tokenization, Pos tagging, Lemmatization & similarity algorithms to understand the similarity score.













Output of this function is list of one relevance value for every parsed resume as either High, Moderate or Low

**Education**

Degree Level

We created function using tokenization, pos tagging and List comparisons to match keywords to find Degree Level of a Resume.

Output of this function is list of one-degree value for every parsed resume as either Masters, Bachelor or other

College Tier

During our Domain discussion with HR team, we found the Tier of the College of a candidate is an important feature while providing weightage. We created function using tokenization, pos tagging and matched the extracted college information with a predefined College Tier Lists.

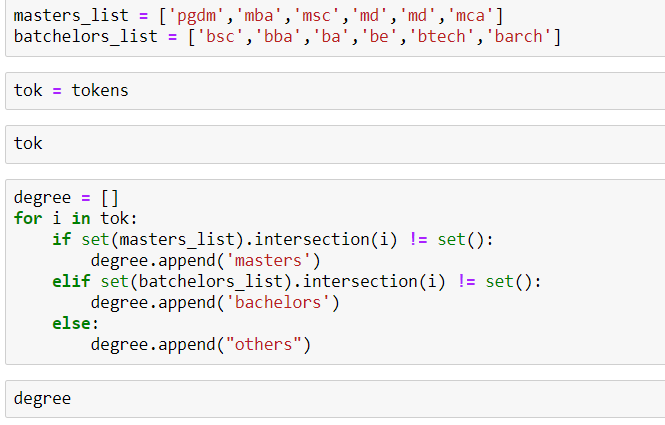
Output of this function is list of one Tier value for every parsed resume as either Tier 1, Tier 2 or other.

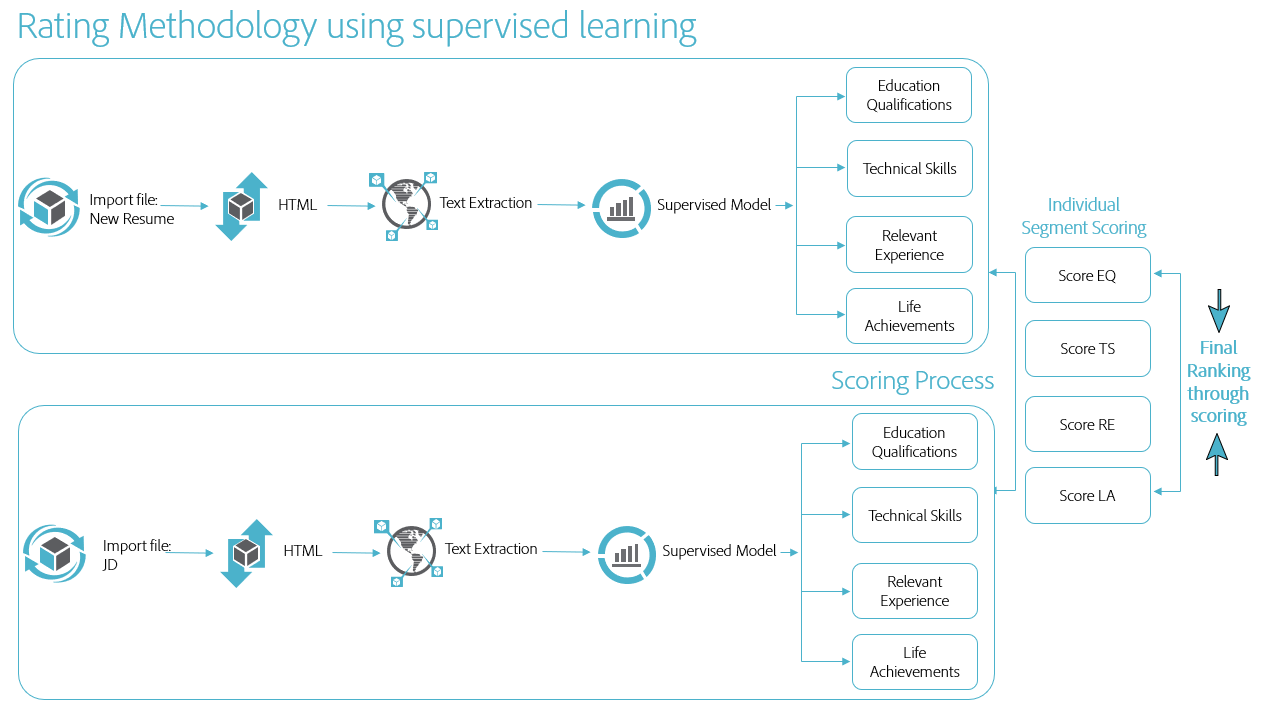










Further, we worked on individual segments, to perform machine learning algorithms. Each segment is different in characteristic and will be treated accordingly. Refer to the framework for the ML techniques to be used for each structured segment.

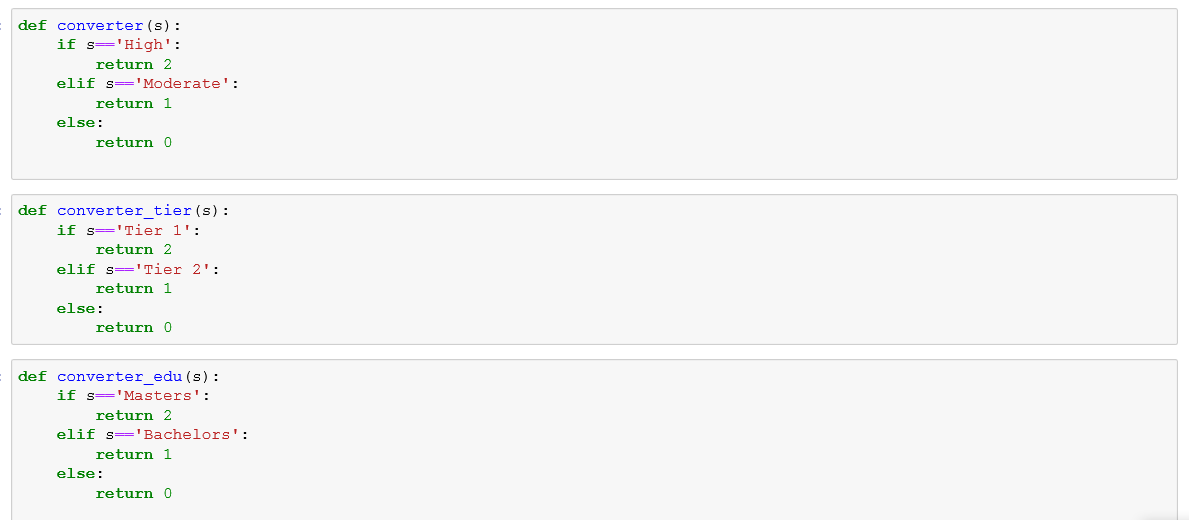
Once the five features with its value are extracted, we convert all the list into data frame.

Supervised model to classify a Resume

Last Stage of framework is to build a Supervised model and then use this model to predict whether a resume is Accepted or Rejected.

We used decision Tree Classifier to build our Model. For training purposes, 36 of the resumes out of 48 of total resumes were labeled manually as Accepted or Rejected.

Code for Supervised Model

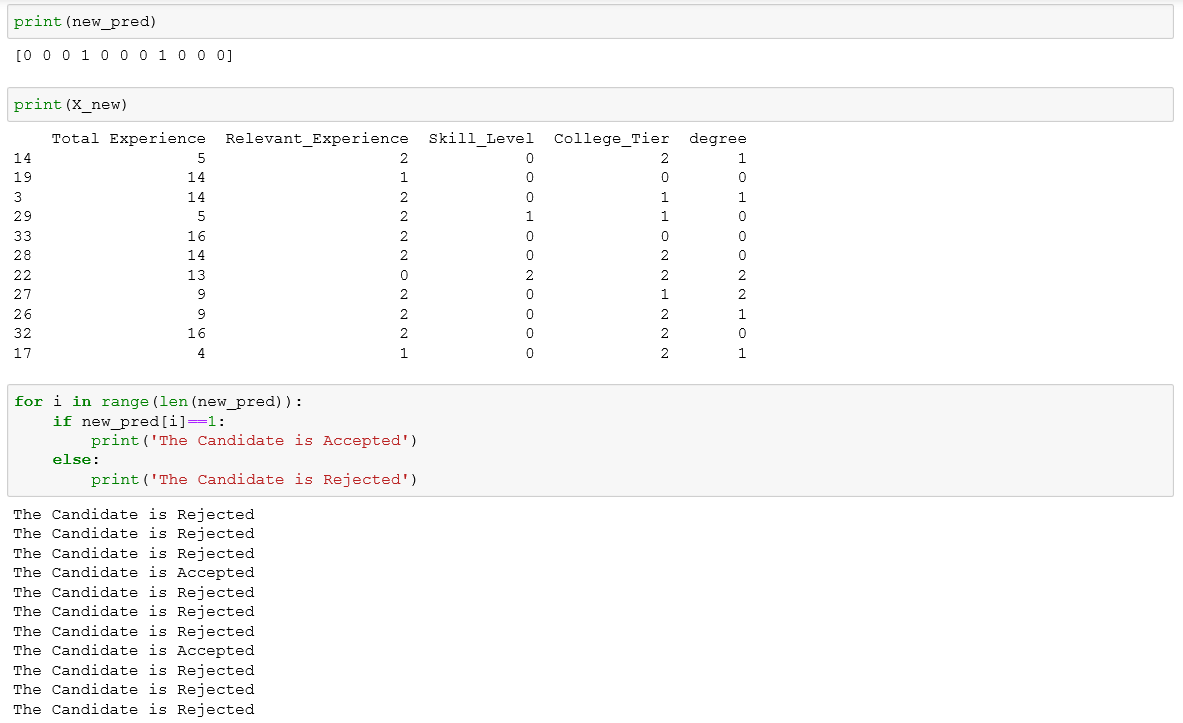




Code for Fitting the model



Output



**Conclusion/Recommendations**

* Process Impact: - A manual process of screening a candidate through 100s of resume would be automated which will be very efficient and less time consuming. We have started with structuring the framework and because we had very few resumes to train on, accuracy will be low at start but while more and more data will be fed, accuracy could be improved.
* Cost Impact: - We four people have spent almost 3 months in defining the problem, creating a framework, deciding to go with and model training. If there is a team of 4 people in house working on same project, most probably they can achieve more efficiency and in house people would have access to real data. Initially it can be costly but once team is well trained and start working on various Text analytics projects, this project will a starting point and it will help company in long term. So, I would say this project can incur some cost but completely justified.
* People need to open to change and adopt screening process through computers. It will still need manual intervention for some time until model start providing accuracy beyond threshold but until then, a feedback loop needs to be created at organization which will input the data back to system and model for training.
* This is a machine learning model which need data and feedback loop for better accuracy and prediction. It is an iterative process where results are fed back with responses and based on that model is calibrated. Initially frequent calibrations would be needed and with course of time, this frequency can be quarterly or biannually.

Further Work: - This model has been created on limited set of data so more data will be needed. Further, based on results feedback loop need to be created for proper calibration of model and to improve the accuracy.

References

* Notes from classroom and various projects we have done during our course.
* Books like Fundamentals of text mining etc

