**Modelling Approach**

1. Hypothesis & Programing language used
   * The problem statement of the project was to predict the sub department of a given user. The training dataset mainly consisted of text information. Hence to build the model I needed to create some relevant features, few of them undertaking in this model were :
     + Relevant Degree
     + Relevant Certification
     + Relevant Skills
     + Relevant Skills Match
     + Industry
     + Sector
     + Work Exp.

Programing language: Python

1. Data Quality
   * Overview of the dataset
   * Data Quality of the dataset
   * Viewing the distinct values & distribution for categorical and continuous columns

Refer to the code: **Code\** **Data Analysis.ipynb** for details

1. Data Cleaning : This section broadly covers :
   * Create a list of educational and certification
   * Create a list of stop words
   * Create features : Relevant Degree , Relevant Certification , Industry , Sector using fuzzy logic and regular expression
   * Imputing missing values for Work Exp , Industry & Sector
   * Standardizing the user skill columns values using sklearn library TfidfVectorizer & fuzzy logic

Refer to the code: **Code\Data\_Cleaning.py** for details and output of this code is: *Data\Topic\_Modelling\topic\_modelling.csv*

1. Topic Model
   * Topic modelling is performed for each sub function. The objective was to find out key topics for each sub function and match it with user skill column based on fuzzy logic at threshold of 70% (this can vary) to find out whether the mentioned skill is relevant and by what percent.
   * Intuitively 25 topics was built for each sub function with top 40 words.

Refer to the code: **Code\Topic\_Models.py** for details

1. Model Build
   * Used H2o library to build the model
   * I have only used dataset where the relevant skills is equal to one for modelling work, the entire dataset is stored in *Modelling\_Dataset\Modelling\_Dataset\_All\_Columns\_V1.csv* file. The dataset used for modelling is *Modelling\_Dataset\ Modelling\_DatasetV1.csv*
   * I have used gradient boosting machine for building the model.
   * The validation strategy includes
     + Creating train ,test, validation dataset with a split of 70% train, 20% test and 10% validation
     + The problem was multiclass prediction hence I have used confusion matrix to access it performance for each dataset

Refer to **Code\Predictive Model.ipynb** notebook for details regarding the following:

* Model Convergence
* Confusion Matrix for train , test and validation dataset
* Feature Importance

1. Future Scope
   * More relevant feature engineering
   * More tuning & data cleaning could be done while doing topic modelling
   * Hyper parameter tuning and trying other algorithm like random forest, logistic regression or deep learning etc.