

## **Idea/Approach Details**

- Ministry Category: AICTE (Education and Skills)
- Problem Statement: Prediction of Admission & Jobs in Engineering & Technology with respect to demographic locations
- Problem Code: #AICTE1
- Team Leader Name: Ranjan Debnath
- College Code: 1-3324198361 (Netaji Subhash Engineering College, Kolkata)

## **Prototype**

The given problem can be divided into two parts viz. prediction of admission and job as well. Demographics of a candidate can be quantified in terms of several attributes like Age, Gender, Annual Household etc. If we have previous data on these we can proceed with building predictive models.

A general pipeline of a predictive model can be as follows:

Training data as input -> Necessary preprocessing -> Normalization if needed -> Fitting this data to a model (for example Support Vector Machine) -> Training the model with the data -> Fine tuning the parameters of the model -> Validating the model -> Testing the model with unseen data.

Data here is crucial, we will need sufficient amount of data for the purpose, if we lag behind with it, it can be compensated by the virtue of Generative Adversarial Networks which are capable of generating data from a similar distribution. Once we have sufficient data we fit this data to different predictive models (for example Logistic Regression, GBM, Artificial Neural Networks etc.) and see which model is performing the best within a reasonable time.

In the admission prediction part the dataset will include some technical details of previously admitted candidates in different colleges (like JEE rank, HS score etc.) and after training the model it will predict whether a new candidate is eligible for admission. During the evaluation of a particular model the error rate can be minimized with optimization algorithms like Linear Optimization, Gradient Descent, Factorization Machine etc.

Similar process will be followed for prediction of jobs also.

(P.S.: Training, Fitting etc. these terms correspond to the fundamentals of Machine Learning.)

## Technology Stack (For a web application primarily)

- Front end: HTML, CSS, JS
- Back end: Python
- Server: Apache Web Server
- RDBMS: PostgreSQL
- Preferable REST Client: Postman
- Cross Platform: Yes

## Idea / Approach details

### **Use Case:**

The system will have an UI that will take several details (based on demography) of a candidate from the user. (Assumption: The details will be according to the attributes of a previously crafted dataset as mentioned in the Solution section). In case any detail is missing that missing value will be dealt with suitable statistical technique (such as Imputation). Upon entering the details the system will yield the prediction of his/her admission.

The system will also allow the users to upload a collection of records (preferably in .csv format) (.csv because of its light-weight) and predict if the records are eligible for admission or not.

**The system can also show the possible reason(s) if the candidate is not fit for admission. For this, some technical details about the candidate will be needed and we need to maintain an existing dataset containing similar set of features for this analysis.**

Same process goes for the prediction of Jobs also. (The dataset will be different though)

**In the job prediction part the system will not only show the probable percentage of the candidate to land into a job but it will also predict the technical skills to be acquired by the candidate and how much those skills will increase the percentage.**

### **Showstopper:**

Data is very vital for this particular problem statement. If we fall short on it, we can incorporate the power of GANs for generating data coming from a similar distribution. The use of GANs makes it quite niche. Besides, we will be developing web services (REST preferably) in order to build our web application. So, scaling it and extending it to various hybrid and native apps will not be an issue.