Python Basics

A quick introduction to,

- Python syntax
- Variable Assignment
- Numbers
- Data Types
- Variables & Operators
- Functions
- Flow Controls
- Conditional Constructs
- Working with External Libraries

Working with External Libraries

- Imports
- Operator overloading
- Survival tips for venturing into the world of external libraries

Python Packages

NUMPY:

- Array
- Array Manipulations
- Functions
- Numerical Operations
- Indexing & Slicing
- Append & Concatenate

Pandas:

Creating, Reading and Writing

Indexing, Selecting & Assigning

Summary Functions and Maps

Grouping and Sorting

Data Types and Missing Values

Renaming and Combining

Introduction to Data Visualization

Data Visualization with Matplotlib & Seaborn

Line Charts

Bar Charts

Heatmaps

Scatter Plots

Area Chart

Pie Chart

Distributions

Box Plot

Choosing Plot Types and Custom Styles

Data Wrangling Techniques

Introduction to Data preprocessing

Importing the Dataset

Character Encodings

Handling Missing Values

Inconsistent Data Entry

Parsing Dates

Working with categorical Data

Splitting the data into Train and Test set

Outlier Analysis

Feature Scaling

Supervised Learning – Regression

Introduction to Regression

Linear Regression

Multi Linear Regression

Polynomial Regression

Decision Tree Regressor

Random Forest Regressor

Supervised Learning - Classification

Introduction to Classification

Logistic Regression

Decision Tree Classification

Random Forest Classification

K-nearest Neighbors

Naïve-Bayes

Support Vector Machine

XGboost

ANN Using Tensorflow & Keras

Introduction to Tensoflow & Keras

Introduction to ANN

Perfom Regression & Classification using ANN

Model Evaluation Metrics

Regression Evaluation Metrics:

- MAE
- MSE
- R Squared
- RMSE
- Classification metrics
- Confusion Metrics
- Accuracy
- Precision
- Recall F1 Score
- AUC ROC Curves

Hyper-parameter Optimization

Handling Imbalanced Data

Oversampling

Undersampling

Ensembling Techniques

SMOTE

Hyper-parameter tuning

Grid Search

Randomize Search

Unsupervised Learning

Introduction to Clustering

K-Means Clustering

Hierarchical Clustering

Clustering use cases

Build & Deploy ML Application

Introduction to different modes of deployment

Working with Flask Framework

Building application with flask framework

Integrating Machine Learning model with web application

Project Description

- Campus recruitment is a strategy for sourcing, engaging and hiring young talent for internship and entry-level positions.

- College recruiting is typically a tactic for medium- to large-sized companies with high-volume recruiting needs, but can range from small efforts (like working with university career centers to source potential candidates) to large-scale operations (like visiting a wide array of colleges and attending recruiting events throughout the spring and fall semester).

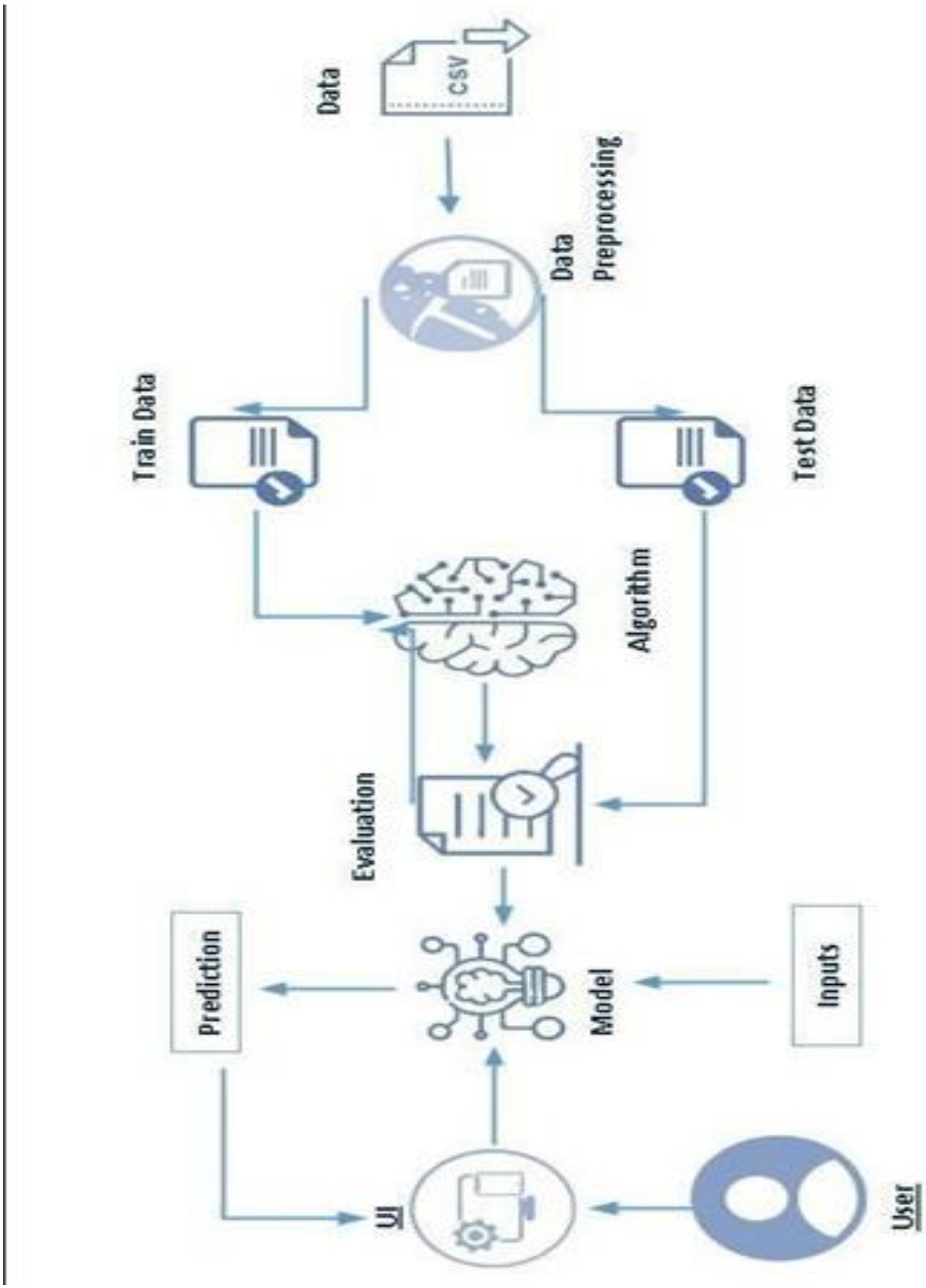
- Campus recruitment often involves working with university career services centers and attending career fairs to meet in-person with college students and recent graduates.

- Our solution revolves around the placement season of a Business School in India. Where it has various factors on candidates getting hired such as work experience,exam percentage etc.

- Finally it contains the status of recruitment and remuneration details.

- We will be using algorithms such as KNN, SVM and ANN. We will train and test the data with these algorithms. From this the best model is selected and saved in .pkl format. We will be doing flask integration and IBM deployment.

Technical Architecture:



Project flow:

- User interacts with the UI to enter the input.
- Entered input is analyzed by the model which is integrated.
- Once model analyzes the input the prediction is showcased on the UI.

To accomplish this, we have to complete all the activities listed below,

Data collection

- Collect the dataset or create the dataset

Visualizing and analyzing data

- Univariate analysis
- Bivariate analysis
- Multivariate analysis
- Descriptive analysis

Data pre-processing

- Checking for null values
- Handling outlier
- Handling categorical data
- Splitting data into train and test

Model building

- Import the model building libraries
- Initializing the model
- Training and testing the model
- Evaluating performance of model
- Save the model

Application Building

- Create an HTML file
- Build python code

Project Flow

Milestone 1: Define Problem / Problem Understanding

- **Activity 1: Specify the business problem**

  Refer Project Description

- **Activity 2: Business requirements**

  The business requirements for a project aimed at "Identifying Patterns and Trends in Campus Placement Data using Machine Learning" would likelyinclude the following:

- Access to campus placement data: The project would requireaccess to data on student performance, qualifications, and job placement outcomes. This data would need to be collected,
  cleaned, and prepared for analysis.

- Machine learning expertise: The project would require individuals with expertise in machine learning, data science andstatistical analysis to develop and implement the algorithms andmodels needed to analyze the data.

- Data storage and management: The project would require a robust and secure data storage and management system to storeand organize the large amounts of data used in the analysis.

- Infrastructure for model deployment: The project would require infrastructure for deploying the models and algorithms developed, including hardware, software, and cloud-based resources.

- **Activity 3: Literature Survey**

  - There have been several studies that have used machine learningtechniques to identify patterns and trends in campus placement data.
  - One study by authors P. K. Rajesh and Dr. G. R. Suresh, published in the International Journal of Computer Science and Mobile Computing in 2015, used k-means clustering and decision trees to analyze campus placement data and identify patterns that could be used to predict placement outcomes.
  - Another study by authors V.V. Kulkarni and K.S. Patil, publishedin the International Journal of Engineering Research and Technology in 2012, used decision tree and neural network algorithms to analyze campus placement data and identify factorsthat influence student placement.

  - A study by authors S.S. Bhosale, S.S. Raut, and D.S. Kulkarni, published in the International Journal of Emerging Research inManagement & Technology in 2013, used decision tree and Naive Bayes algorithms to analyze campus placement data andpredict student placement outcomes.

- A study by authors S.S. Bhosale, S.S. Raut, and D.S. Kulkarni, published in the International Journal of Emerging Research inManagement & Technology in 2013, used decision tree and Naive Bayes algorithms to analyze campus placement data andpredict student placement outcomes.
- In general, these studies found that machine learning techniques were effective at identifying patterns and trends in campus placement data, and could be used to predict student placementoutcomes with high accuracy.
- It's important to note that all these studies are quite old now and you might find more recent studies and new techniques whichcan be useful for your project.

Conclusion

The algorithms of machine learning we have discussed are can used to find the trend of placement, which will be helpful for university to get more admission in future. We compared the algorithm and find out the accuracy by considering some of attributes of students. Here we used deep neural network classifier with the 1000, 2000, 5000 iteration with 71%, 77%, and 91% of accuracy.