Identifying Patterns and Trends in Campus Placement Data Using Machine Learning

Milestone 1: Define Problem / Problem Understanding

Activity 1: Specify the business problem

Introduction

In this activity, we will be discussing the business problems and goals of this project, as well as the data collection process and our understanding of the project. The business problem we are addressing is related to campus placement, where the company aims to improve the efficiency of student placement. Currently, the placement process is done manually, which can result in errors and mistakes. To address this issue, we will be developing a machine learning model to simplify and accelerate the process of analyzing placement data and ultimately help the company achieve its goal. To better understand the project let's take a closer look at the process of data collection and insights gained from analyzing the data.

Problem Statement

Here the problem is the process of analyzing the data is manual currently, which may lead to errors and mistakes in hiring the students when there is a large amount of data to sort it out the company wants a machine learning model which can analyze large amounts of data without any errors. The reason for the company to develop a machine learning model is to improve the efficiency and accuracy of hiring the students for internship and entry level positions, and to analyze the large dataset without any sort of errors.

Methodology

The methodology adopted for this project involves several steps that are necessary for the development of an efficient and accurate machine learning model for analyzing the campus placement data. The methodology comprises four stages: data preprocessing, model selection, and evaluation.

Data preprocessing means cleaning the gathered data and organizing it into an order. The data has been cleaned by removing the missing values and outliers, and then it has been transformed by normalizing the features to ensure that they have a similar scale. We should prepare it for analysis by transforming the raw data gathered.

Model selection: In this section we are using three models to train them and then test them then finally select the best on from them to analyze a new data set to check if it accurately analyzes it. The three models selected respectively are: K-Nearest Neighbors (KNN), Support Vector Machines (SVM), and Artificial Neural Networks (ANN).

Evaluation: In this final stage we evaluate the selected best machine learning algorithm and test it with several metrics such as accuracy, precision, recall, and F1 score to evaluate the performance of the model.

Data Collection

To train the machine learning model and test it, we need a data set. After cleaning, organizing, and normalizing the raw data gathered from the sources we use it train the models. Here we gathered the data from this website but there are other sources to collect the data too: https://www.kaggle.com/code/neesham/prediction-of-placements/data

The dataset contains information about the placement status of students who graduated from a particular college. It includes various features such as gender, degree specialization, work experience, and salary offered.

The dataset consists of 215 entries with 15 features. The data was collected through a survey conducted by the college, and it represents the placement data for the academic year 2018-19. The data was stored in a CSV file format, and it was preprocessed before using it in the model.

Conclusion

In conclusion, this project aims to develop a machine learning model that can analyze campus placement data efficiently and accurately, while also improving the placement process for the company. The methodology used for the project involves four stages: data preprocessing, model selection, and evaluation. In addition, we gathered data from an external source and cleaned it to prepare it for analysis.

In the upcoming activities we discuss about: Business requirements, Literature survey, Social or Business Impact of the project.

Overall, the successful completion of this project will provide the company with an efficient and accurate way to analyze campus placement data, leading to better decision-making and improved placement outcomes.

Activity 2: Business requirements

Business requirements:

Business requirements refer to the specific needs and objectives that a business must meet in order to achieve its goals. These requirements can encompass a wide range of areas, including product development, marketing, sales, finance, and operations.

To develop effective business requirements, it is important to first clearly define the business's overall goals and objectives. This might include increasing revenue, expanding into new markets, improving customer satisfaction, or reducing costs.

Once the business's goals are established, it is necessary to identify the specific requirements that must be met in order to achieve those goals. This might involve conducting market research, analyzing financial data, or gathering feedback from customers.

Business requirements should be documented in a clear and concise manner, and should be regularly reviewed and updated as needed. Effective communication and collaboration between different departments and stakeholders is also essential to ensuring that business requirements are met in a timely and efficient manner.Ultimately, effective business requirements can help a business to stay competitive and achieve long-term success by guiding strategic decision-making and ensuring that resources are allocated in the most effective way possible.

BENEFITS:

- Alignment: Business requirements help ensure that everyone involved in the project understands the goals and objectives of the business, and is aligned in working towards them.
- Clarity: Well-defined business requirements provide clarity on what needs to be achieved, how it needs to be achieved, and what resources are required to achieve it.
- Efficiency: Having clear business requirements helps teams to work more efficiently by reducing the risk of miscommunication, errors, and rework.
- Accountability: Business requirements provide a basis for measuring progress and holding team members accountable for delivering the expected results.
- Prioritization: Business requirements help teams to prioritize their work based on the business needs and objectives, ensuring that they are delivering the most valuable outcomes first.
- Validation: Business requirements provide a framework for validating that the project or initiative has achieved its intended goals and objectives, and that the deliverables meet the business needs.

DESIGN:

To design a placement business requirements, you will need to consider various aspects of the business, including its goals, target audience, staffing needs, budget, and marketing strategies. Here are some steps you can follow to design effective placement business requirements:

•Define the business goals: Start by identifying the business's primary goals, such as placing qualified candidates in suitable job positions, building a reputation for reliability and efficiency, increasing revenue, and expanding the business's network.

•Identify the target audience: Determine the type of job seekers your placement business will cater to, such as entry-level or experienced candidates, specific industries, or job positions.

•Staffing needs: Determine the number and type of staff you will need, including recruiters, account managers, administrative staff, and marketing professionals.

•Budget: Set a budget for your placement business, including costs for staffing, office space, marketing, and technology.

•Technology requirements: Determine the software and tools you will need to manage job postings, applicant tracking, candidate screening, and communication with clients and candidates.

•Marketing strategies: Develop marketing strategies to attract potential clients, such as social media marketing, email marketing, and advertising in industry-specific publications.

•Legal requirements: Research and comply with legal requirements for your placement business, such as labor laws, data protection laws, and anti-discrimination laws.

•Measure success: Establish key performance indicators (KPIs) to measure the success of your placement business, such as placement rate, client satisfaction, and revenue growth.

•Job posting database: The business also needs to maintain a database of job postings from various employers. This database should include details such as the job title, job description, qualifications required, and other relevant information.

•Matching algorithm: The placement business should have an algorithm that matches job seekers with appropriate job postings based on their skills, qualifications, and preferences.

•Resume building tool: A placement business may also offer a resume building tool to help job seekers create effective resumes that highlight their skills and experience.

•Communication tools: The business needs to provide communication tools for job seekers to interact with employers and vice versa. This may include email, messaging, or video conferencing tools.

•Payment system: The business needs to have a payment system in place to charge employers for job postings and other services offered.

•Privacy and data security: The placement business should have robust privacy and data security measures in place to protect the personal information of job seekers and employers.

•Legal compliance: The business needs to comply with relevant labor laws and regulations in the regions where it operates.

•Customer service: The business should provide excellent customer service to job seekers and employers, including prompt responses to queries and complaints.

•Marketing and advertising: The placement business needs to invest in marketing and advertising to attract both job seekers and employers to its platform.

Activity 3: Literature Survey

Campus placement is an important aspect of the hiring process for many organizations. It is a process where companies visit educational institutions to recruit fresh talent for various roles in their organization. A literature survey on campus placement can provide insight into various aspects of the process, such as its effectiveness, factors affecting it, and strategies for improvement.

One study by Dhankhar and Kaushik (2017) explored the effectiveness of campus placement in India. The study found that campus placement is an effective method for recruitment as it saves time and money for companies and provides job opportunities for students. However, the study also found that the placement process can be improved by increasing industry-academia collaboration, providing better training to students, and incorporating modern technology in the recruitment process.

Another study by Singh and Gupta (2019) investigated the factors affecting the success of campus placement. The study found that factors such as communication skills, technical knowledge, academic performance, and personality traits play a crucial role in the recruitment process. The study suggested that educational institutions should focus on developing these skills in students to enhance their employability.

A study by Giri and Kumar (2018) explored the role of placement cells in improving the campus placement process. The study found that placement cells play a crucial role in bridging the gap between students and companies. The study suggested that placement cells should focus on building relationships with companies, providing training to students, and improving the overall placement process.

Lastly, a study by Kumar and Dhamija (2020) investigated the impact of COVID-19 on the campus placement process. The study found that the pandemic has led to a shift in the recruitment process from traditional on-campus recruitment to virtual recruitment. The study suggested that

educational institutions should adapt to this change by providing online training and preparing students for virtual interviews.

Overall, the literature survey highlights the importance of campus placement in the recruitment process and suggests various strategies for improving its effectiveness. Educational institutions should focus on developing students' skills and building relationships with companies to enhance their placement rates. Furthermore, the pandemic has brought about significant changes in the recruitment process, and educational institutions should adapt to these changes to prepare students for the future.

Activity 4: Social or Business Impact

Social impact

The first dimension of a sustainable business is its performance relative to societies and social justice, often referred to as social impact. While there is no easy solution for reducing social costs while improving corporate performance and profitability, social impact should not be overlooked. The social impact of a business's operations is viewed both internally and externally and ensures that the business's entire operations across the supply chain are socially responsible and ethical.

6 benefits of social impact for businesses

1: Builds trust with the community

2: Creates meaningful change

3: Keeps employees engaged

4:Ensures sustainability

5: Attracts loyal customers

6: Tells a story people care about

The sustainable business is not only expected to treat its employees in a responsible manner but also ensure that it is engaged with suppliers that share similar values. That is, a sustainable business is also concerned for the labor practices and working conditions of companies within its supply chain to ensure that the supplies and products it purchases were produced responsibly and ethically. Sustainable businesses will make reasonable efforts to ensure they are not purchasing from suppliers engaged in the use of sweatshops, child labor, or other human rights abuses. In some cases, businesses have worked diligently with suppliers to correct these problems, while in other cases businesses have chosen to change suppliers.

Milestone 2: Data Collection & Preparation:

Introduction:

Machine Learning (ML) algorithms depend heavily on data, and the quality of the data has a direct impact on the accuracy of the ML model. The data collection and preparation phase of the ML project is a critical step that involves collecting data from reliable sources and cleaning and pre-processing the data to make it suitable for training the ML model. In this milestone, we will cover the activities involved in data collection and preparation.

Activity 1: Collect the dataset

The first step in any machine learning project is to collect the necessary data. There are many popular sources for collecting data, such as Kaggle.com, the UCI Machine Learning Repository, and more. For this project, we will be using a .csv dataset that we downloaded from Kaggle. The dataset we are using is related to the prediction of job placement outcomes for students. The data includes various features such as gender, degree specialization, work experience, and more.

Activity 1.1: Importing the libraries

Before we can begin working with the dataset, we need to import the necessary libraries. In this activity, we will be importing the pandas library, which is a popular library for data manipulation and analysis in Python.

Activity 1.2: Read the Dataset

After importing the necessary libraries, we can now read the dataset using the read_csv() function from pandas. We provide the directory of the csv file as a parameter to the function. Once we have read the dataset, we can explore the data to understand its structure and contents.

Activity 2: Data Preparation

After understanding the structure of the dataset, we need to preprocess the data before we can use it to train our machine learning model. The downloaded dataset may contain missing data, categorical data, or other issues that need to be addressed before we can use it for training. In this activity, we will be handling missing data and categorical data.

Activity 2.1: Handling missing values

Missing values are a common issue in real-world datasets. In this activity, we will find the shape of our dataset using the df.shape method and determine the data type using the df.info() function. We will then handle any missing values in the dataset using various techniques such as dropping rows or columns with missing values or filling in the missing values with appropriate values.

Activity 2.2: Handling outliers

Outliers are data points that lie far from the typical range of values in a dataset. They can skew our analysis and negatively impact the performance of our machine learning model. In this activity, we will detect any outliers in the dataset using statistical methods such as box plots, scatter plots, or Z-scores. We will then handle any outliers using various techniques such as removing them from the dataset or replacing them with more appropriate values.

Activity 2.3: Handling Categorical Values

Categorical data refers to data that takes on discrete values from a limited set of categories. In our dataset, we have categorical data that needs to be converted into numerical data so that it can be used in our machine learning model. In this activity, we will use encoding techniques such as replacement to convert the categorical features into numerical features. This will allow us to include these features in our machine learning model and improve its performance.

Milestone 3: Exploratory Data Analysis

Data analysis is an important step in any machine learning project. It involves understanding the dataset, identifying patterns and relationships, and making decisions about how to prepare the data for modeling. Exploratory data analysis (EDA) is the process of analyzing and visualizing data to gain insights and identify patterns. In this milestone, we will be performing EDA on our dataset.

Activity 1: Visual Analysis

Visual analysis is the process of using visual representations, such as charts, plots, and graphs, to explore and understand data. It is a way to quickly identify patterns, trends, and outliers in the data, which can help to gain insights and make informed decisions.

Activity 1.1: Univariate Analysis

Univariate analysis involves analyzing one variable at a time. In this activity, we will be using two different plots to analyze the distribution of our target variable "status" and "salary".

The first plot is a distplot, which is a histogram with a line on top that shows the density estimate. The distplot gives us an idea about the distribution of our data. We can see that the "status" variable has two possible values: "placed" and "not placed". The histogram shows that the majority of students are placed, while a small percentage are not. The "salary" variable, on the other hand, has a skewed distribution with a long tail towards higher salaries.

The second plot is a countplot, which is a bar chart that shows the number of occurrences of each category in a categorical variable. We use this plot to analyze the distribution of the categorical variables "gender", "ssc_b", "hsc_b", "hsc_s", "degree_t", and "workex". From the countplot, we can see that there are more males than females in our dataset, and that most of the students have completed their secondary and higher secondary education from central boards. The majority of students have a science background in higher secondary, while a smaller percentage have a commerce or arts background. The most common degree is commerce and management, followed by science and technology. A majority of students do not have work experience.

Activity 1.2: Bivariate Analysis

Bivariate analysis involves analyzing the relationship between two variables. In this activity, we will be using a countplot to analyze the relationship between "status" and "degree_t". We will also be using a boxplot to analyze the relationship between "status" and "salary".

The countplot shows the number of students who are placed and not placed for each type of degree. We can see that the highest percentage of placed students have a degree in commerce and management, while the highest percentage of not placed students have a degree in science and technology.

The boxplot shows the distribution of salaries for students who are placed and not placed. We can see that the median salary of placed students is higher than the median salary of not placed students.

Activity 1.3: Multivariate Analysis

Multivariate analysis involves analyzing the relationship between three or more variables. In this activity, we will be using a swarmplot to analyze the relationship between "status", "degree_t", and “specialization”.

The swarmplot shows the relationship between "status", "degree_t", and "specialization". Each dot represents a student, and the color of the dot indicates whether the student is placed or not.

We can see that most of the students who are placed have a degree in commerce and management, and have specialized in marketing and finance.

Activity 2: Scaling the data

In this activity, we perform scaling on our dataset. Scaling is a critical step as data that measures in different ranges can lead to misleading results in prediction models such as KNN and logistic regression, which rely on distance-based methods and gradient descent concepts. Scaling ensures that each feature is treated equally, and the algorithm is not biased towards a particular feature.

In this project, we use the StandardScaler method from the sklearn library to scale the data. This method scales the data to have a mean of 0 and a standard deviation of 1, thus standardizing the range of values. We pass the training dataset through the StandardScaler.fit_transform() method to calculate the scaling parameters and transform the data, and then transform the test dataset with the StandardScaler.transform() method using the same scaling parameters.

Activity 3: Splitting the data into train and test

In this activity, we split our dataset into training and testing sets. The purpose of this step is to train the model on a portion of the data and test its performance on a separate, unseen portion of the data. This allows us to evaluate how well the model generalizes to new data.

We first separate the target variable (placement status) from the rest of the data using the pandas drop() function. The resulting dataset with the target variable removed is assigned to the variable X, and the target variable is assigned to the variable y.

We then use the train_test_split() function from the sklearn library to split the data into training and testing sets. This function takes in the X and y variables and randomly splits them into training and testing sets based on a specified test size. We set the test size to 0.2, meaning that 20% of the data will be reserved for testing, and the remaining 80% will be used for training. We also set the random state to 42 to ensure reproducibility of the results.

The train_test_split() function returns four variables: X_train, X_test, y_train, and y_test. These variables represent the training and testing sets of the independent variables (‘X’) and dependent variable (‘y’). The training sets are used to train the model, while the testing sets are used to evaluate the performance of the model.

In conclusion, the data collection and preparation and exploratory data analysis are critical steps in the machine learning workflow. In the data collection and preparation stage, we collected the dataset, read it into our program, and pre-processed it to ensure that it is suitable for model training. In the exploratory data analysis stage, we visualized and analyzed the data to gain insights and understand the relationships between the variables. We also scaled the data to

ensure that each feature is treated equally and split the data into training and testing sets to evaluate the performance of our model. By following these steps, we can ensure that our model is trained on the right data and that we can trust its predictions on new, unseen data

Milestone 4: Model Building

The first step in building the model is to train it using multiple algorithms. For this project, we are applying four different classification algorithms, and the best model will be saved based on its performance. The algorithms we will be using are SVM, KNN, and an artificial neural network.

Activity 1: Training the model in multiple algorithms

We will begin by training the model using the SVM algorithm. To do this, we will create a function called Support vector machine, which will take the training and test data as parameters. Inside the function, we will initialize the SVMClassifier algorithm and pass the training data to the model using the .fit() function. We will then use the .predict() function to make predictions on the test data and save the results in a new variable. To evaluate the performance of the model, we will create a confusion matrix and classification report.

Next, we will use the KNN algorithm to train the model. To do this, we will create a function called KNN, which will take the training and test data as parameters. Inside the function, we will initialize the KNeighborsClassifier algorithm and pass the training data to the model using the .fit() function. We will then use the .predict() function to make predictions on the test data and save the results in a new variable. To evaluate the performance of the model, we will create a confusion matrix and classification report.

Finally, we will use an artificial neural network to train the model. This algorithm is more complex than the previous two and will require a different approach. We will first create a function to preprocess the data, which will involve scaling the data and converting the categorical variables to numerical data. We will then create a neural network model using the Keras library. This will involve creating a sequential model and adding layers to it. We will then compile the model and fit it to the training data. Once the model has been trained, we will use it to make predictions on the test data and evaluate its performance using a confusion matrix and classification report.

Once all three models have been trained and evaluated, we will select the best model based on its performance and save it for use in the next milestone. The best model will be the one with the highest accuracy and lowest error rate.

Overall, the goal of Milestone 4 is to build a model that accurately predicts the outcome variable based on the input variables. This involves training the model using multiple algorithms, evaluating its performance using confusion matrices and classification reports, and selecting the best model based on its accuracy and error rate. The model will be saved for use in the next milestone, where it will be used to make predictions on new data.

Milestone 5: Model Deployment

In this milestone, we will be deploying the model we built in the previous milestones and integrating it into a web application. This will involve saving the best model, building HTML pages, building server-side scripts, and running the web application.

Activity 1: Save the best model

The first step in model deployment is to save the best model we built based on its performance. We can do this by selecting the model with the highest performance and saving its weights and configuration. This is important because it avoids the need to retrain the model every time it is needed and allows us to use it in the future.

Activity 2: Integrate with Web Framework

In this section, we will be building a web application that is integrated with the model we built. A UI is provided for the user where they have to enter the values for predictions. The entered values are given to the saved model, and the prediction is showcased on the UI. This section has the following tasks:

Activity 2.1: Building HTML Pages

For this project, we will create three HTML files and save them in the templates folder:

index.html: This will be the main page of the application.

index1.html: This page will contain the form to enter the values for prediction.

secondpage.html: This page will display the prediction output.

Activity 2.2: Building HTML Pages (part 2)

We will write the following code to fetch the details from the user using the <form> tag. A submit button is provided at the end which navigates to the prediction page upon clicking. The code for secondpage.html includes a section that provides the output on the screen.

Activity 3: Build Python code

The final step in the deployment process is to build the server-side script in Python. This will involve importing the necessary libraries, rendering the HTML pages, and defining the main function of the application.

Activity 3.1: Import the libraries

First, we need to load the saved model. Importing the Flask module in the project is mandatory. An object of Flask class is our WSGI application. Flask constructor takes the name of the current module (name) as an argument.

Activity 3.2: Render HTML page

We will be using a declared constructor to route to the HTML page we created earlier. In the above example, '/' URL is bound with the home.html function. Hence, when the home page of the web server is opened in the browser, the HTML page will be rendered. Whenever you enter the values from the HTML page, the values can be retrieved using the POST method. The below code retrieves the value from UI.

We are routing our app to the predict() function. This function retrieves all the values from the HTML page using the POST request. These values are stored in an array, which is then passed to the model.predict() function. This function returns the prediction, which will be rendered to the text that we mentioned in the submit.html page earlier.

Activity 3.3: Main Function

To run the web application, we need to open the Anaconda prompt from the start menu, navigate to the folder where the Python script is, and type the “python app.py” command. Then, we can navigate to the localhost where we can view the web page.

Clicking on the predict button from the top right corner will open the form where we can enter the inputs, click on the submit button, and see the result/prediction on the web. We can copy the link from the prompt and paste it in Chrome. A web page opens up where we need to enter the values into the fields provided to get our final prediction.

Overall, this milestone involves deploying the model we built and integrating it with a web application. It requires knowledge of web frameworks like Flask, HTML, and server-side scripting in Python. With this deployment, we can provide a user-friendly interface for people to interact with our machine learning model and get predictions in real-time.