Task 3.1:

Data preprocessing is a step in the data mining and data analysis that takes raw data and transforms it into a format that can be understood by computers and machine learning. Data is in the form of text, images, video, pdf. These may sometimes contain errors and inconsistencies. So these things have to be cleaned and formatted and data preprocessing is used.

Data preprocessing steps:

1.Data quality assessment

In this step the data is assessed to get the overview of data regarding the project. There are many errors that one can find in this step like-

Mismatched data types: When you collect data from many different sources, it may come to you in different formats. For simplicity, you need to convert them into a single format. For example, if part of your analysis involves family income from multiple countries, you'll have to convert each income amount into a single currency.

Mixed data values: Different sources use different descriptions for example, woman or female. These values should all be made uniform.

Data outliers: Outliers can have a huge impact on data analysis results. For example, if you're averaging test scores, and one student didn't respond to any of the questions, their 0% could greatly hamper the results.

The step is necessary as we get an overall idea of our data and many inconsistencies are found in this.

2**.**Data Integration

Data Integration is one of the data preprocessing steps that are used to merge the data present in multiple sources into a single larger data store like a data warehouse. We might run into some issues while adopting Data Integration as one of the Data Preprocessing steps like the data can be present in different formats, and attributes that might cause difficulty in data integration.

Data Integration is needed especially when we are aiming to solve a real-world scenario.

3. Data cleaning

Data cleaning is the process of correcting, repairing, or removing incorrect or irrelevant data from a data set. Dating cleaning is the most important step of preprocessing because it will ensure that your data is ready to go for your downstream needs. There are many ways to correct for missing data:

Ignore the tuples: A tuple is an ordered list or sequence of numbers or entities. If multiple values are missing within tuples, you may simply discard the tuples with that missing information. This is only recommended for large data sets when a few ignored tuples won't harm further analysis.

Manually fill in missing data: This can be tedious but is necessary when working with smaller data sets.

Noisy data: It is the Data that includes unnecessary data points, irrelevant data, and data that are more difficult to group. For cleaning the Noisy data:

Binning: Binning sorts data of a wide data set into smaller groups of more similar data. Income, for example, could be grouped: $35,000-$50,000, $50,000-$75,000, etc.

Regression: Regression is a statistical technique used to model and analyse the relationship between a dependent variable (also known as the target or outcome variable) and one or more independent variables (also called predictors or features). The primary goal of regression analysis is to predict the value of the dependent variable based on the values of the independent variables.

Clustering: Clustering algorithms are used to properly group data, so that it can be analysed with like data.

If you're working with text data, for example, you should consider to Remove URLs, symbols, emojis, etc., that aren't relevant to your analysis and translate all text into the language you'll be working in.

4. Data transformation

With data cleaning, we've already begun to modify our data, but data transformation will begin the process of turning the data into the proper format you'll need for analysis and other downstream processes.

Aggregation: Data aggregation combines all of your data in a uniform format.

Normalization: Normalization scales your data into a regularized range so that you can compare it more accurately. For example, if you're comparing employee loss or gain within several companies (some with just a dozen employees and some with 200+), you'll have to scale them within a specified range, like -1.0 to 1.0 or 0.0 to 1.0.

Feature selection: Feature selection is the process of deciding which variables (features, characteristics, categories, etc.) are most important to your analysis. These features will be used to train ML models.

Discretization: It pools data into smaller intervals. For example, when calculating average daily exercise, rather than using the exact minutes and seconds, you could join together data to fall into 0-15 minutes, 15-30, etc.

5. Data Reduction

The size of the dataset in a data warehouse can be too large to be handled by data analysis and data mining algorithms. The solution is to obtain a reduced representation of the dataset that is much smaller in volume but produces the same quality of analytical results.

Attribute selection: Similar to discretization, attribute selection can fit your data into smaller pools. It, essentially, combines tags or features, so that tags like male/female and professor could be combined into male professor/female professor.

Dimensionality reduction: This reduces the amount of data used to help facilitate analysis and downstream processes. Algorithms like K-nearest neighbours use pattern recognition to combine similar data and make it more manageable.

Task 3.2

Diffusion Models

Diffusion Models are generative models, meaning that they are used to generate data similar to the data on which they are trained. Diffusion models are advanced machine learning algorithms that uniquely generate high-quality data by progressively adding noise to a dataset and then learning to reverse this process. The main idea here is to add random noise to data and then undo the process to get the original data distribution from the noisy data.

Diffusion models work in a dual-phase mechanism: They first train a neural network to introduce noise into the dataset and then methodically reverse this process.

Firstly, the data is pre-processed for a better output.

Then the forward diffusion process begins with the image that undergoes a series of reversible, additions of noise to the images following a Markov chain. Markov chain is a chain of events where the current time step depends only on the previous time step so there are no cross dependencies between time steps that do not immediately follow each other. So, this helps in tracking the noise adding to be reversed later on. This process continuous till the image consists only of noise (could be 100 or 1000 steps).

The noise that is added in diffusion models is Gaussian Noise. It is a noise that has a probability distribution of a Gaussian distribution (Given different mean and variance the bell shape of the graph stays the same). Adding this to an image means changing the values of pixels of that image.

Reverse process means recovering the value of pixels so that the resulting image will resemble the original image. This is achieved by using neural networks. To do that we input the image to a convolutional neural network (CNN) and then the network produces the image in the previous step. CNNs can extract features from noisy images at each time step to understand the underlying structure and content. The type of convolutional network used is the U-Net (called that because of its shape). Through the convolutions, it makes a small representation of the image and then samples it back to the original dimensions, this way the input and output dimensions of the network have the same size.

To generate a new image, the process starts with random noise and iteratively applies the reverse steps to gradually remove the noise, eventually producing a clean image that looks like it could belong to the training dataset.

This method allows diffusion models to generate highly diverse and high-quality images.

Clustering

The task of grouping data points based on their similarity with each other is called Clustering. It's useful in dealing with unlabeled and unstructured data. This method is defined under the branch of Unsupervised Learning, a branch of machine learning that deals with unlabeled data. Unsupervised learning algorithms are tasked with finding patterns and relationships within the data without any prior knowledge of the data's meaning.

Types of clustering:

Centroid-based

Centroid-based clustering works on the closeness of the data points to the chosen central value. The datasets are divided into a given number of clusters, and a vector of values references every cluster. The input data variable is compared to the vector value and enters the cluster with minimal difference.

K-means clustering is a popular centroid-based clustering algorithm used to partition a dataset into k distinct, non-overlapping clusters. The goal is to divide the data points in such a way that each point belongs to the cluster with the nearest mean (centroid), thereby minimizing the variance within each cluster.

Pre-defining the number of clusters at the initial stage is the most crucial yet most complicated stage for the clustering approach. Despite the drawback, it is a vastly used clustering approach for surfacing and optimizing large datasets.

These groups of clustering methods iteratively measure the distance between the clusters and the characteristic centroids using various distance metrics. These are either Euclidian distance, Manhattan Distance or Minkowski Distance.

Hierarchical Clustering

Hierarchical clustering builds a hierarchy of clusters either by merging smaller clusters into larger ones (agglomerative approach) or by splitting larger clusters into smaller ones (divisive approach). The result is a tree-like diagram called a dendrogram, which shows the relationships among clusters. In the agglomerative approach, each data point is its cluster. At each step, merge the two closest clusters based on a chosen distance metric based on

Single Linkage: the distance between the two clusters is the shortest distance between points in those two clusters.

Complete Linkage: the distance between the two clusters is the farthest distance between points in those two clusters.

Average Linkage: the distance between the two clusters is the average distance of every point in the cluster with every point in another cluster.

The divisive approach in hierarchical clustering, also known as top-down clustering, starts with the entire dataset as a single cluster and recursively splits it into smaller clusters. This process continues until each data point is in its own cluster or until a stopping criterion is met, such as a specified number of clusters or a minimum cluster size.

Reinforcement Learning

Reinforcement Learning is a feedback-based Machine learning technique in which an agent learns to behave in an environment by performing the actions and seeing the results of actions. For each good action, the agent gets positive feedback, and for each bad action, the agent gets negative feedback or a penalty. In Reinforcement Learning, the agent learns automatically using feedback without any labeled data, unlike supervised learning. Since there is no labeled data, the agent is bound to learn by its experience only.

The agent interacts with the environment and explores it by itself. The primary goal of an agent in reinforcement learning is to improve performance by getting the maximum positive rewards.

The agent continues doing these three things (take action, change state/remain in the same state, and get feedback), and by doing these actions, he learns and explores the environment.

The agent learns what actions lead to positive feedback or rewards and what actions lead to negative feedback penalties. As a positive reward, the agent gets a positive point, and as a penalty, it gets a negative point.

The agent learns with the process of hit and trial, and based on the experience, it learns to perform the task in a better way. Hence, we can say that "Reinforcement learning is a type of machine learning method where an intelligent agent (computer program) interacts with the environment and learns to act within that."

It is a core part of Artificial intelligence, and all AI agent works on the concept of reinforcement learning. Here we do not need to pre-program the agent, as it learns from its own experience without any human intervention.

Robotic Control

In robotics, RL can teach robots to perform tasks such as walking, picking up objects, or navigating through environments. The robot learns by receiving rewards for successful actions and penalties for mistakes. For instance, a robot learning to walk might get rewards for maintaining balance and moving forward, and penalties for falling.

Autonomous Vehicles

RL is used to train self-driving cars to navigate roads, follow traffic rules, and avoid obstacles. The car’s RL agent receives rewards for safe driving behaviours (e.g., staying within lane boundaries) and penalties for unsafe behaviours (e.g., collisions). Over time, the agent learns to drive in a way that maximizes safety and efficiency.

Game Playing (e.g., Chess, Go)

RL agents can be trained to play games by learning from simulations or historical game data. For example, AlphaGo, developed by DeepMind, used RL to master the game of Go by playing millions of games against itself and learning from each game’s outcomes. The agent learns strategies that maximize its chance of winning based on rewards given for winning or losing.