EXPLORING STRUCTURE OF DATA

The data structure used for machine learning is quite similar to other software development fields where it is often used. Machine Learning is a subset of artificial intelligence that includes various complex algorithms to solve mathematical problems to a great extent. Data structure helps to build and understand these complex problems. Understanding the data structure also helps you to build ML models and algorithms in a much more efficient way than other ML professionals.

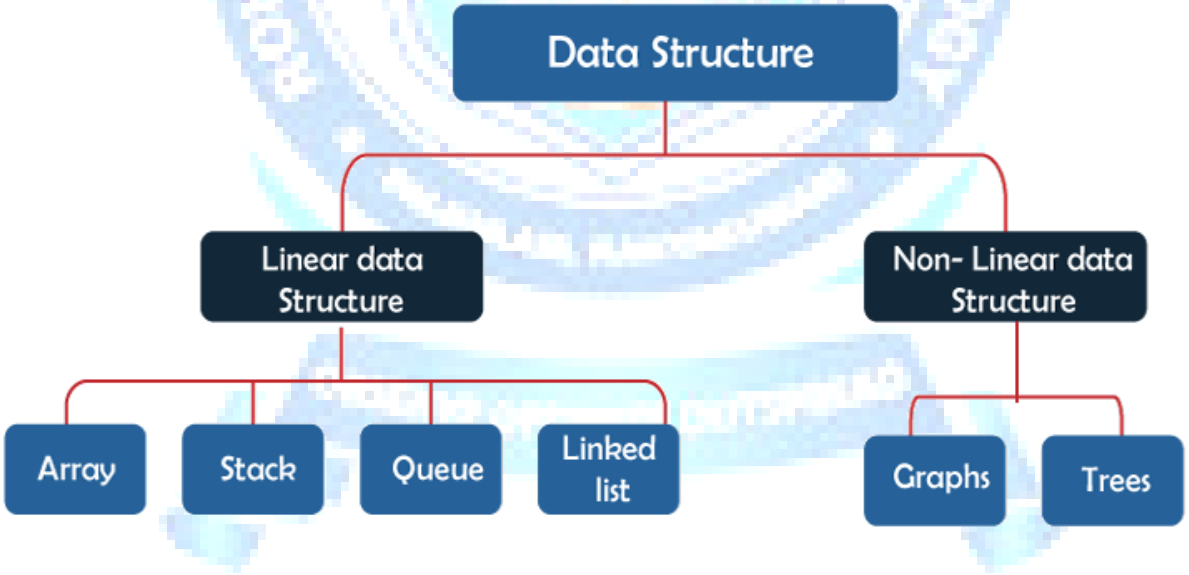
What is Data Structure?

The data structure is defined as the basic building block of computer programming that helps us to organize, manage and store data for efficient search and retrieval. In other words, the data structure is the collection of data type 'values' which are stored and organized in such a way that it allows for efficient access and modification.

Types of Data Structure

The data structure is the ordered sequence of data, and it tells the compiler how a programmer is using the data such as **Integer, String, Boolean, etc.**

There are two different types of data structures: Linear and Non-linear data structures.



1. Linear Data structure:

The linear data structure is a special type of data structure that helps to organize and manage data in a specific order where the elements are attached adjacently.

There are mainly 4 types of linear data structure as follows:

Array:

An array is one of the most basic and common data structures used in Machine Learning. It is also used in linear algebra to solve complex mathematical problems. You will use arrays constantly in machine learning, whether it's:

- To convert the column of a data frame into a list format in pre-processing analysis
- To order the frequency of words present in datasets.
- Using a list of tokenized words to begin clustering topics.
- In word embedding, by creating multi-dimensional matrices.

An array contains index numbers to represent an element starting from 0. The lowest index is arr[0] and corresponds to the first element.

Let's take an example of a Python array used in machine learning. Although the Python array is quite different from than array in other programming languages, the Python list is more popular as it includes the flexibility of data types and their length. If anyone is using Python in ML algorithms, then it's better to kick your journey from array initially.

Python Array method:

Method	Description
Append()	It is used to add an element at the end of the list.
Clear()	It is used to remove/clear all elements in the list.
Copy()	It returns a copy of the list.
Count()	It returns the count or total available element with an integer value.
Extend()	It is used to add the element of a list to the end of the current list.
Index()	It returns the index of the first element with the specified value.
Insert()	It is used to add an element at a specific position using an index number.
Pop()	It is used to remove an element from a specified position using an index number.
Remove()	Used to remove the elements with specified values.
Reverse()	Used to show list in reverse order
Sort()	Used to sort the list in an array.

Stacks:

Stacks are based on the concept of LIFO (Last in First out) or FILO (First In Last Out). *It is used for binary classification in deep learning.* Although stacks are easy to learn and implement in ML models but having a good grasp can help in many computer science aspects such as parsing grammar, etc.

Stacks enable the **undo** and **redo** buttons on your computer as they function similar to a stack of blog content. There is no sense in adding a blog at the bottom of the stack.

However, we can only check the most recent one that has been added. Addition and removal occur at the top of the stack.

Linked List:

A linked list is the type of collection having several separately allocated nodes. Or in other words, *a list is the type of collection of data elements that consist of a value and pointer that point to the next node in the list.*

In a linked list, insertion and deletion are constant time operations and are very efficient, but accessing a value is slow and often requires scanning. So, a linked list is very significant for a dynamic array where the shifting of elements is required. Although insertion of an element can be done at the head, middle or tail position, it is relatively cost consuming. However, linked lists are easy to splice together and split apart. Also, the list can be converted to a fixed-length array for fast access.

Queue:

A Queue is defined as the "FIFO" (first in, first out). It is useful to predict a queuing scenario in real-time programs, such as people waiting in line to withdraw cash in the bank. Hence, the queue is significant in a program where multiple lists of codes need to be processed.

The queue data structure can be used to record the split time of a car in F1 racing.

2. Non-linear Data Structures

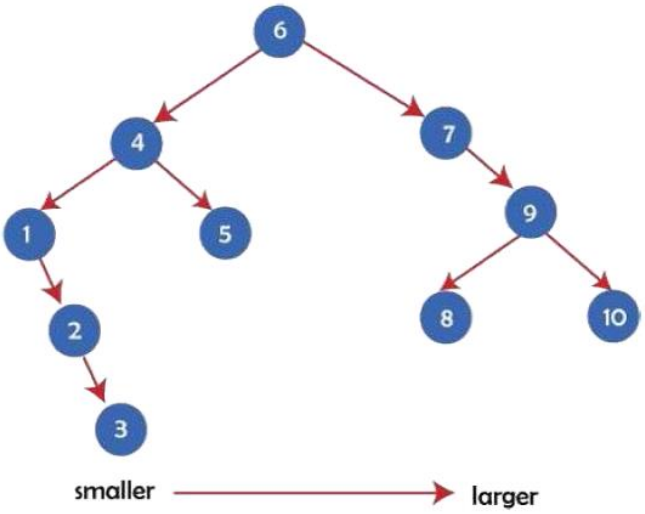
As the name suggests, in Non-linear data structures, elements are not arranged in any sequence. All the elements are arranged and linked with each other in a hierarchal manner, where one element can be linked with one or more elements.

1) Trees

Binary Tree:

The concept of a binary tree is very much similar to a linked list, but the only difference of nodes and their pointers. In a linked list, each node contains a data value with a pointer that points to the next node in the list, whereas; *in a binary tree, each node has two pointers to subsequent nodes instead of just one.*

Binary trees are sorted, so insertion and deletion operations can be easily done with O(log N) time complexity. Similar to the linked list, a binary tree can also be converted to an array on the basis of tree sorting.



In a binary tree, there are some child and parent nodes shown in the above image. Where the value of the left child node is always less than the value of the parent node while the value of the right-side child nodes is always more than the parent node. Hence, in a binary tree structure, data sorting is done automatically, which makes insertion and deletion efficient.

2) Graphs

A graph data structure is also very much useful in machine learning for link prediction. Graphs are directed or undirected concepts with nodes and ordered or unordered pairs. Hence, you must have good exposure to the graph data structure for machine learning and deep learning.

3) Maps

Maps are the popular data structure in the programming world, which are mostly useful for minimizing the run-time algorithms and fast searching the data. It stores data in the form of (key, value) pair, where the key must be unique; however, the value can be duplicated. Each key corresponds to or maps a value; hence it is named a Map.

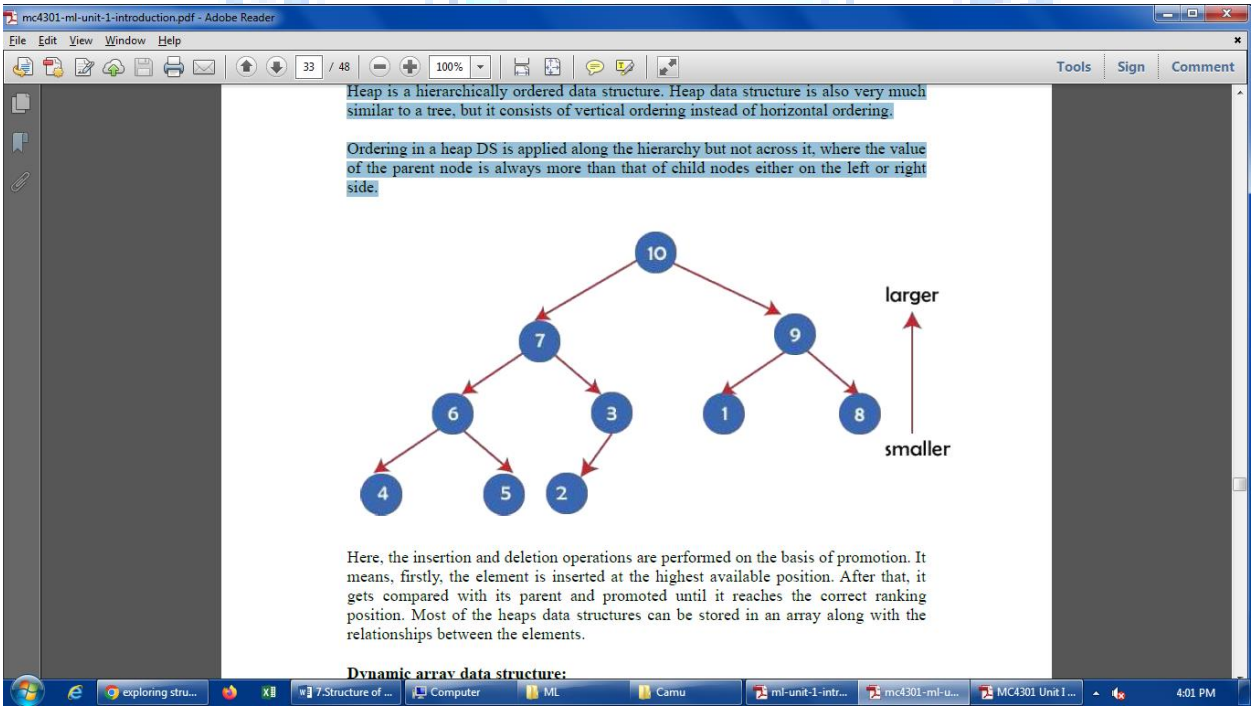
In different programming languages, core libraries have built-in maps or, rather, HashMaps with different names for each implementation.

- **In Java: Maps**
- **In Python: Dictionaries**
- **C++: hash_map, unordered_map, etc.**

Python Dictionaries are very useful in machine learning and data science as various functions and algorithms return the dictionary as an output. Dictionaries are also much used for implementing sparse matrices, which is very common in Machine Learning.

4) Heap data structure:

Heap is a hierarchically ordered data structure. Heap data structure is also very much similar to a tree, but it consists of vertical ordering instead of horizontal ordering. Ordering in a heap DS is applied along the hierarchy but not across it, where the value of the parent node is always more than that of child nodes either on the left or right side.



Here, the insertion and deletion operations are performed on the basis of promotion. It means, firstly, the element is inserted at the highest available position. After that, it gets compared with its parent and promoted until it reaches the correct ranking position. Most of the heaps data structures can be stored in an array along with the relationships between the elements.

Dynamic array data structure:

This is one of the most important types of data structure used in linear algebra to solve 1-D, 2-D, 3-D as well as 4-D arrays for matrix arithmetic. Further, it requires good exposure to Python libraries such as **Python NumPy** for programming in deep learning.

How is Data Structure used in Machine Learning?

For a Machine learning professional, apart from knowledge of machine learning skills, it is required to have mastery of data structure and algorithms.

When we use machine learning for solving a problem, we need to evaluate the model performance, i.e., which model is fastest and requires the smallest amount of space and resources with accuracy. Moreover, if a model is built using algorithms, comparing and contrasting two algorithms to determine the best for the job is crucial to the machine learning professional. For such cases, skills in data structures become important for ML professionals.

With the knowledge of data structure and algorithms with ML, we can answer the following questions easily:

- How much memory is required to execute?
- How long will it take to run?
- With the business case on hand, which algorithm will offer the best0020performance?