Section 18.1 - Artificial Intelligence

Layer 6: High-Order Language

Syllabus Content Section 18: Artificial Intelligence

S18.1.1 Show understanding of how graphs can be used to aid Artificial Intelligence (AI)

- Purpose and structure of a graph
- Use A* and Dijkstra's algorithms to perform searches on a graph
- Candidates will not be required to write algorithms to set up, access, or perform searches on graphs

---

Graph:

- A graph is a collection of nodes or vertices between which there can be edges.
- Each node has a name
- An edge can have an associated label which is a numerical value

The use of graph to aid Artificial Intelligence:

- Artificial neural networks can be represented by graphs
- Graphs provide structure for relationships // graphs provide relationships between nodes
- AI problem can be defined as finding a path in a graph
- Graph may be analysed/injected by a range of algorithms
  - A* / Dijkstra's algorithm
  - used in machine learning
- Back propagation of errors / regressions

> Dijkstra's Algorithm

Dijkstra's algorithm is an algorithm for finding the shortest paths between nodes in a graph, which may represent, for example, road networks.

- Limitations
  - **A lack of heuristics**

  Dijkstra's algorithm has no notion of the overall shortest direction to the end goal, so it will actually spend a lot of time searching in completely the wrong direction if the routes in the wrong direction are shorter than the route in the correct direction. It will find the shortest route in the end but it will waste a lot of time. In small networks this isn't a problem but when you have massive networks (like road networks or the internet) then it will result in massive inefficiencies.

- **Negative Weighted Costs**

  On physical networks with physical distances you can't have negative weights, but one some networks where you are calculating costs you might have negative costs for a particular leg. Dijkstra's cant handle these negative costs.

- **Directed networks**

  Dijkstra's algorithm doesn't always work best when there are directed networks (such as motorways that only run in one direction.

> A* Algorithm

- One of the biggest problems with Dijkstra's algorithm is that it is that it has the potential to be inefficient when searching for the shortest path, because it just looks for the next shortest leg.
- A* is an informed search algorithm, or a best-first search, meaning that it is formulated in terms of weighted graphs: starting from a specific starting node of a graph, it aims to find a path to the given goal node having the smallest cost (least distance travelled, shortest time, etc.). It does this by maintaining a tree of paths originating at the start node and extending those paths one edge at a time until its termination criterion is satisfied.
- At each iteration of its main loop, A* needs to determine which of its paths to extend. It does so based on the cost of the path and an estimate of the cost required to extend the path all the way to the goal.

S18.1.2 Show understanding of how artificial neural networks have helped with machine learning

---

State the reasons for having multiple hidden layers in an artificial neural network

- Enable deep learning to take place
- Where the problem you are trying to solve has a higher level of complexity it requires more layers
- Allow neural network to learn and make decisions on its own
- To improve the accuracy of the result

Explain how artificial neural network enable machine learning

- Artifical neural networks are intended to replicate the way human brain work
- Weights / values are assigned for each connection between nodes
- The data are input at the input layer, then passed into the system
- They are analyzed at each subsequent hidden layer where characteristics are extracted / output are calculated
- This process of training / learning is repeated mant times to achieve optimum outputs // reinforcement learning takes place

- Decisions can be made without being specifically programmed
- The output layer provides the result
- Back propagation of errors will be used to correct any errors that have been made

S18.1.3 Show understanding of Deep Learning, Machine Learning and Reinforcement Learning and the reasons for using these methods.

- Understand machine learning categories, including supervised learning, unsupervised learning

---

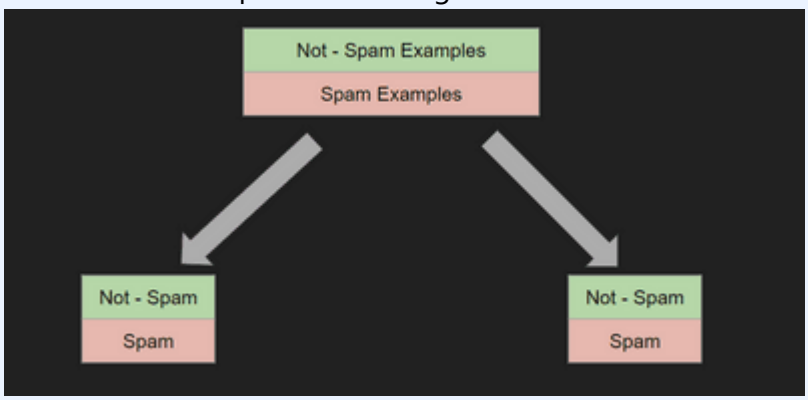
Deep Learning & Neural Networks

Artificial Neural Networks are computational models and inspire by the human brain. Many of the recent advancements have been made in the field of Artificial Intelligence, including Voice Recognition, Image Recognition, Robotics using Artificial Neural Networks.

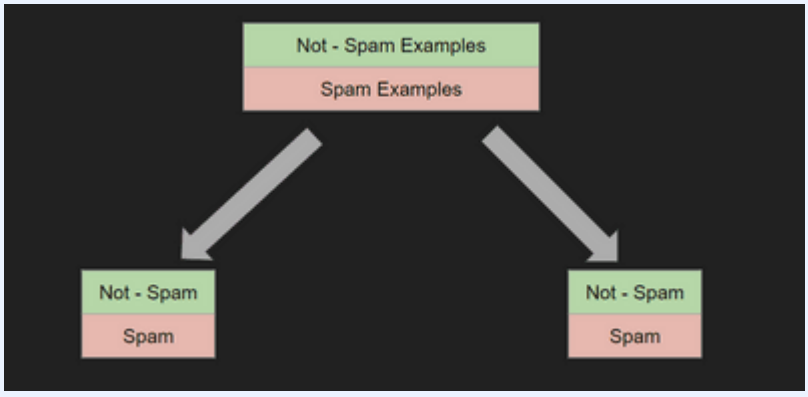
Supervised Learning

This is where you feed the machine learning algorithm labelled training data. The labels contain the expected outcome for that data. The machine used the labels and training data to train the model.

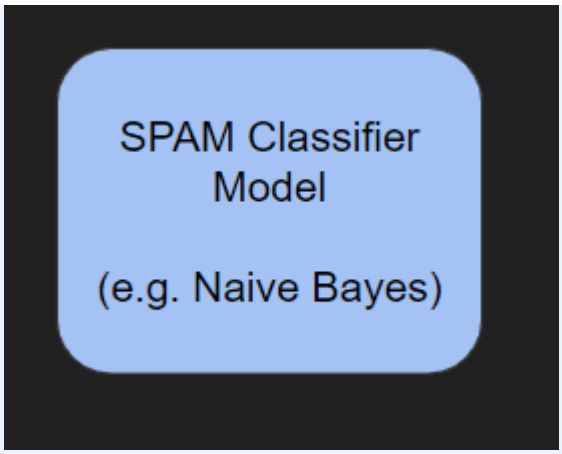
Labelled data is split into training and test data



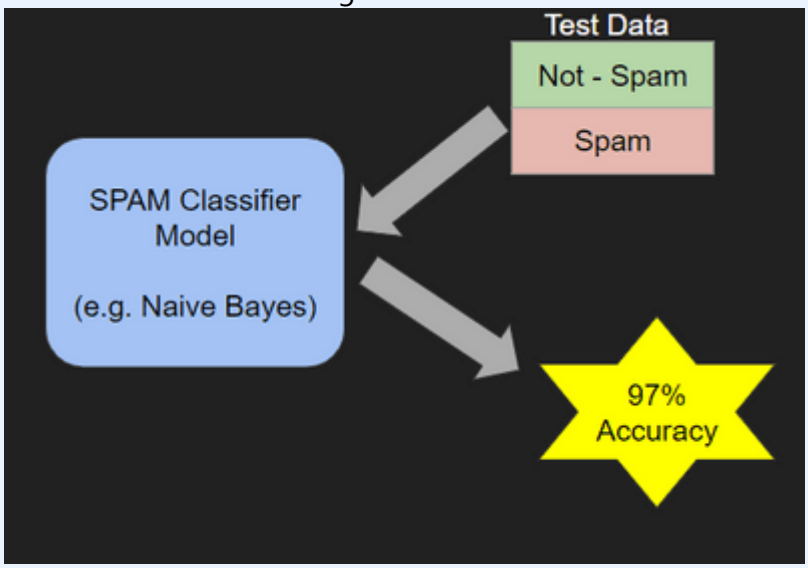
Labelled data is split into training and test data



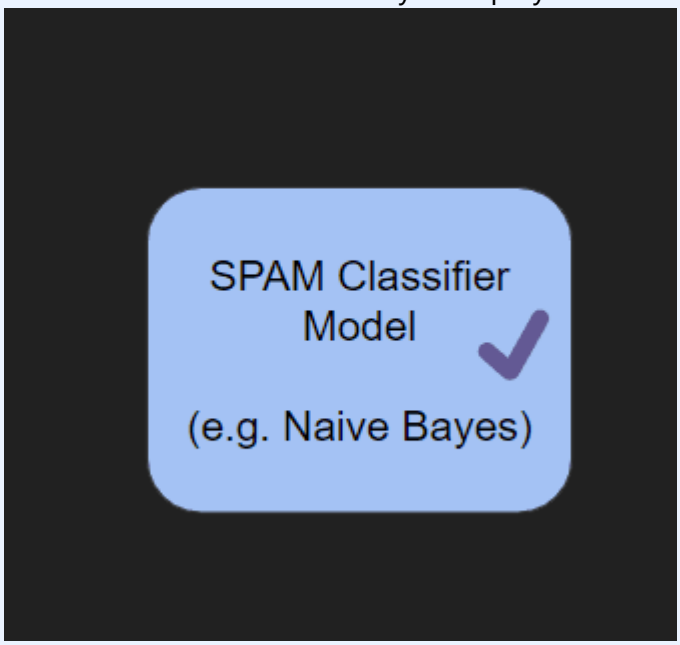
Trained model is ready to test



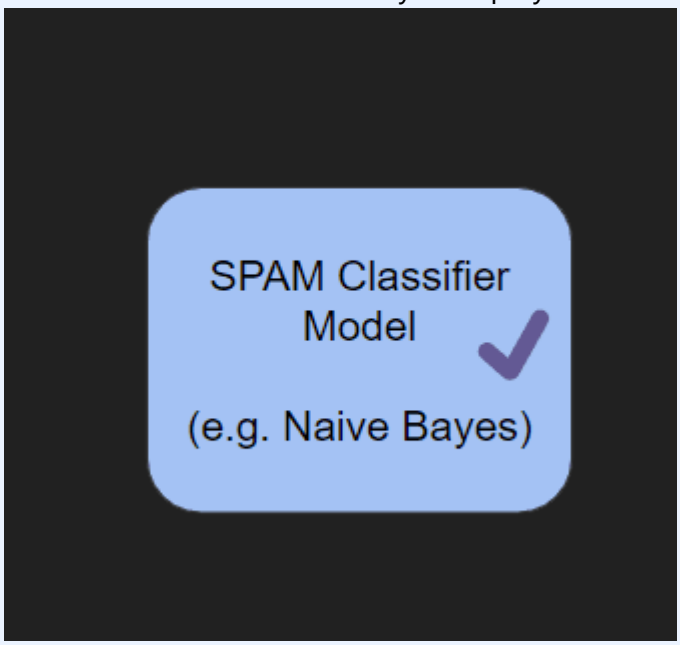
Model is then tested using the test data



The tested model is then ready to deploy



The tested model is then ready to deploy



> Unsupervised Learning

Machine learning algorithm is trained on unlabelled data and is left to cluster data itself. Certain hyper-parameters may be set (such as how many clusters to form) but the process is generally unstructured.

- Useful for categorising many different objects
- Identifying hidden trends or patterns
- Anomaly detection(e.g. fraudulent transactions, spotting skin cancer, crime detection)

S18.1.4 Show understanding of back propagation of errors and regression methods in machine learning

---

> Disadvantages of using Backpropagation

- The actual performance of backpropagation on a specific problem is dependent on the input data.
- Back propagation algorithm in data mining can be quite sensitive to noisy data
- You need to use the matrix-based approach for backpropagation instead of mini-batch.