DL UNIT – 1

Machine Learning (ML)

ML is a subfield of AI which concerns with developing computational theories of learning and building learning machines.

Computer program is set to learn from experience E with respect to some class of task T and performance measure P, it's performance in task T, which is measured in P improves with experience E.

It is very hard to write a program that solve problems like recognizing a human face because we do not know what program to write because we don't know how our brain does it. Instead of writing a program by hand, it is possible to collect a lot of examples that specify the correct output for a given input.

Machine learning algorithm, then takes these examples and produces the program that does, the job. If we do it right the program will also work for new cases as well as the ones, we trained it on.

Limitations

Machine learning models are often opaque, it makes us difficult to understand why a particular prediction was made.

Machine learning models are often unstable, meaning that they can produce different result when trained on different data set.

Machine learning models are often difficult to customize, it can be hard to change their parameters.

Machine learning models are often expensive to train meaning that it can take a lot of time and computational resource to build a model that is accurate.

Machine learning models are often vulnerable to learning from noise in the data which can lead to inaccurate prediction.

Machine learning models are often sensitive to the distribution of data that can produce different result if the data is sorted in different way.

Deep Learning (DL)

DL is a subset of machine learning, which is predicated on idea of learning from an example.

Deep learning, eliminate some of data pre-processing that is typically involved with machine learning.

In deep learning a computer model learns to perform classification tasks directly from image, text, and sound.

Deep learning models can achieve state of the art accuracy sometimes exceeding human level performance.

Models are trained by using a large set of label data and neural network architecture that contains many layers.

Pros

Deep learning models achieve high accuracy in various tasks due to their ability to learn complex patterns from data.

They can automatically extract relevant features from raw data, eliminating the need for manual feature engineering.

With advancements in hardware and software, deep learning models can scale effectively to handle large datasets and complex problems.

They exhibit adaptability by adjusting to new data and environments, making them suitable for dynamic scenarios.

Pre-trained deep learning models can be fine-tuned for specific tasks, saving time and computational resources.

Cons

Deep learning models require large amounts of labelled data for training, which can be expensive and time-consuming to acquire.

Training deep learning models often demands significant computational resources, including powerful GPUs or TPUs and large amounts of memory.

Complex deep learning architectures are prone to overfitting, memorizing noise or specific patterns in the training data rather than generalizing well to unseen data.

Deep learning models lack explainability, making it difficult to understand how they arrive at their predictions.

Dependence on large datasets for achieving high performance can be a limitation in domains where data is scarce or expensive to collect.

Machine learning	Deep learning
Machine learning is a super set of Deep learning.	Deep learning is a subset of machine learning.
Machine learning algorithms are linear.	Deep learning algorithm are complex and nonlinear.
Machine Learning consist of thousands of data points	Deep learning works on big data so millions of data points.
Not necessary to have costly high-end machines	High end machines and high performing GPU are required.
Machine Learning is evolution of AI.	Deep learning is an evolution of machine learning, basically it is how deep is the machine learning.
Machine Learning is highly used to stay in competition and learn new things.	Deep learning solves complex learning issues.
More human intervention is involved for getting results.	Although they require additional set up time, deep learning algorithm can produce result immediately.
The result of machine learning model is easy to explain.	The results of Deep learning are difficult to explain.
Machine learning models can be used to solve straight forward a little bit challenging issue.	Deep learning models are appropriate for resolving challenging issues.

Artificial Intelligence (AI)

AI focuses on giving machines, cognitive and intellectual capabilities similar to those of humans.

AI aims towards building machines that are capable to think like humans.

It is a subset of data science. Artificial Intelligence uses logic and decision tree.

It is used for development of computerised application that simulate human intelligence and interaction.

Its objective is to maximise the chance of success.

AI encompasses a collection of intelligence concept, including elements of perception, planning and prediction.

It is concerned with knowledge, dissemination, and conscious machine actions.

Co-relation

Artificial Intelligence (AI), Machine Learning (ML), and Deep Learning (DL) are interconnected fields, each building upon the other, but they are not synonymous.

Artificial Intelligence (AI)

AI is the broader concept of machines being able to carry out tasks in a way that we would consider "intelligent."

It encompasses various techniques, methodologies, and algorithms aimed at mimicking human intelligence.

AI can be classified into two categories: Narrow AI (Weak AI) and General AI (Strong AI).

Narrow AI refers to AI systems that are designed and trained for a specific task, such as speech recognition, image recognition, or playing chess.

General AI refers to AI systems that possess the ability to understand, learn, and apply knowledge across a wide range of tasks, exhibiting human-like intelligence.

Machine Learning (ML)

ML is a subset of AI that focuses on the development of algorithms that allow computers to learn from and make predictions or decisions based on data.

ML algorithms learn from data without being explicitly programmed to do so.

ML algorithms can be categorized into three main types: supervised learning, unsupervised learning, and reinforcement learning.

Supervised learning involves learning a mapping from inputs to outputs based on labelled training data.

Unsupervised learning involves learning patterns or representations from unlabelled data.

Reinforcement learning involves learning to make decisions by interacting with an environment and receiving feedback in the form of rewards or penalties.

Deep Learning (DL)

DL is a subfield of ML that focuses on artificial neural networks with multiple layers (deep neural networks).

DL algorithms attempt to model high-level abstractions in data by using multiple processing layers with complex structures, or neural networks, composed of multiple layers of nodes.

DL has been particularly successful in tasks such as image and speech recognition, natural language processing, and playing games.

DL has gained popularity due to its ability to automatically learn features from raw data, eliminating the need for manual feature extraction.

Supervised learning / Unsupervised learning

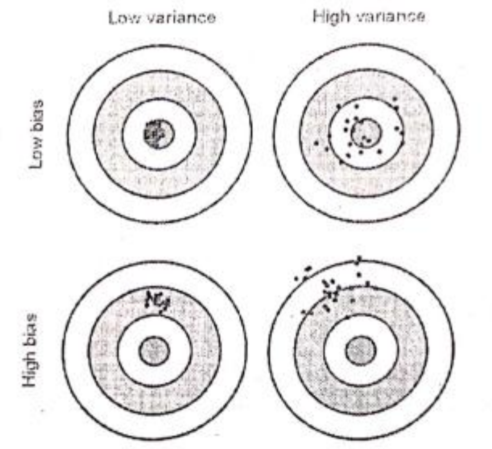
Supervised learning	Unsupervised learning
Supervised learning algorithm are trained using labelled data.	Unsupervised learning algorithm are trained using unlabelled data.
Supervised learning model takes direct feedback to check if it is predicting correct output or not.	Unsupervised learning model does not take any feedback.
Supervise learning model predict the output.	Unsupervised learning model finds the hidden pattern in data.
In supervise learning, input data is provided to the model along with output.	In unsupervised learning, only input data is provided to the model.
The goal of supervisor learning is to train the model so that it can predict the output when given new data.	The goal of unsupervised learning is to find the hidden pattern and useful insight from the unknown data set.

Supervise learning needs supervision to train the model.	Unsupervised learning does not need any supervision to train the model.
Supervisor can be categorised in classification and regression problem.	Unsupervised learning can be classified in clustering and association problems.
Supervise learning can be used for those cases where we know the input as well as corresponding outputs.	Unsupervised learning can be used for those cases where we have only input data and no corresponding output data.
Supervise learning model produces an accurate result.	unsupervised learning model may give less accurate result as compared to supervise learning.
Supervise learning is not close to true artificial intelligence, as in this, we first train the model for each data and then only it can predict the correct output.	Unsupervised learning is closer to artificial intelligence as it learns similarly as a child learns daily routine things by his experiences.
It includes various algorithm such as linear regression, logistic regression, support vector machine, multiclass classification, decision tree, etc.	It includes various algorithm such as clustering, KNN and apriori algorithm.

Bias Variance Trade-off

Bias - Assumption made by model to make function easier to learn. The algorithm error rate on the training set is algorithm bias.

Variance - If u train your model on training data and obtain a very low error and upon changing the data and then training the same previous model you experience high error, this is variance.



Underfitting (High Bias and Low Variance)

A statistical model of a machine learning algorithm is set to have underfitting when it cannot capture the underlying trend of the data.

It usually happens when we have less data to build an accurate model and also when we try to build a linear model with a non-linear dataset.

In such cases, the rules of the machine learning model are too easy and flexible to be applied on such minimal data and therefore the model will probably make a lot of wrong predictions.

Underfitting can be avoided by using more data and also reducing the feature by using feature selection.

Overfitting (High Variance and Low Bias)

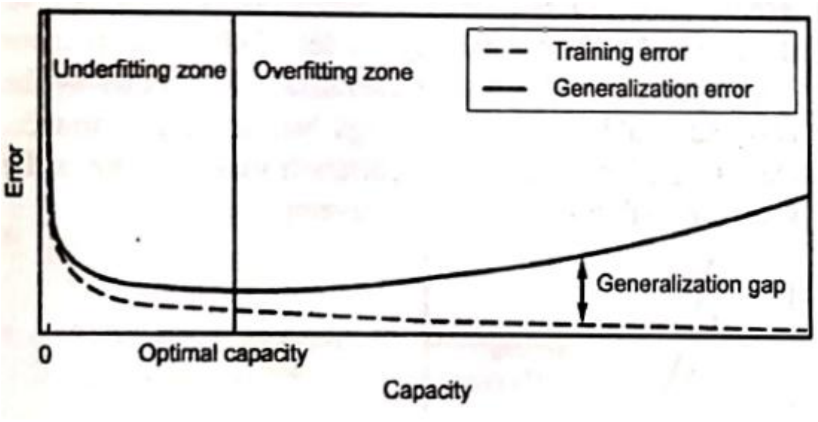
Statistical model is said to be overfitted when we train it with a lot of data.

When model gets trained with so much of data, it starts learning from the noise and inaccurate data entries in the dataset.

Then the model does not categorise the data correctly because of too many details and noise.

The cause of overfitting is the non-parametric and non-linear method because these types of machine learning algorithms have more freedom in building the model based on the dataset and therefore, they can really build unrealistic models.

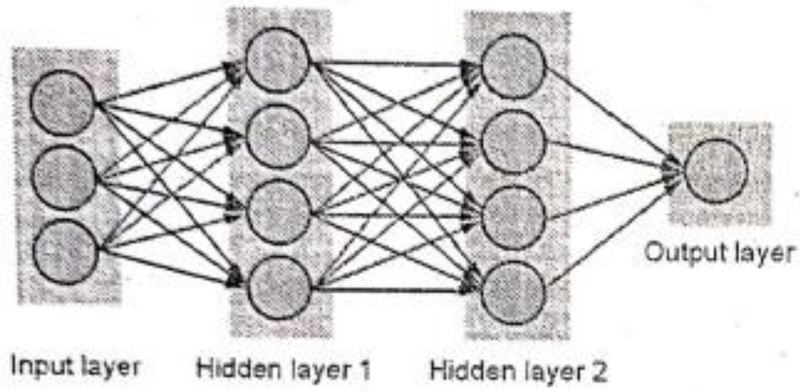
A solution to avoid overfitting is using a linear algorithm if we have linear dataset or using the parameters like the maximal depth if we are using decision tree.



Understanding How Deep Learning Works in Three Figures

In machine learning, we take some data, train a model on that data, and use the trained model to make predictions on new data. The process of training a model can be seen as a learning process where the model is exposed to new, unfamiliar data step-by-step.

At each step, the model makes predictions and gets feedback about how accurate its generated predictions were. This feedback, which is provided in terms of an error, according to some major is used to correct the errors made in prediction.



The **first figure** that can help in understanding deep learning is the architecture of neural network. A neural network is composed of layers of interconnected 'neurons', which are inspired by the structure and function of a biological neuron in the brain.

Each neuron in a layer receive input from previous layer, process it, and send it to the next layer. The input layer receives the raw data and output layer produces the final output of the network. The layers in between are called as 'hidden layers' and they are used to extract features and representations of the data.

The **second figure** that can help in understanding deep learning is the process of training a neural network. In supervise learning, a data with labelled examples is used to train the network.

The network is presented with inputs and the corresponding desired outputs, and its weight and biases are adjusted to minimise the difference between the network predictions and the desired outputs.

This process is repeated for many examples in the dataset and the network gradually learns to make accurate predictions on new, unseen examples.

The **third figure** that may help in understanding deep learning is forward and backward propagation.

It is the process of passing input data through the layers of neural network and computing the output, as well as adjusting the weights of the network in the backward pass by using an optimisation algorithm, like stochastic Gradient Descent (SGD).

This is the process which the network learns from data by minimising the error between the predicted output and the actual output.

Common architectural principles of deep network

<u>Use of multiple layers</u>: Deep networks are composed of multiple layers of interconnecting nodes, which allow them to learn increasingly abstract representation of input data.

<u>Use of non-linear activation functions</u>: Non-linear activation functions, such as ReLU or sigmoid, are used in the nodes of the network to introduce non-linearity into the computation.

<u>Gradient based learning</u>: Deep networks are trained using gradient-based optimisation algorithm, such as stochastic gradient descent, to adjust the weight of the network to minimise the error on the training data.

<u>Use of backpropagation</u>: Backpropagation is a method of training neural networks, which allows the gradient of the error with respect to the weight of the network to be efficiently computed.

<u>Use of dropout</u>: Dropout is a method to prevent overfitting in deep network, which consist of randomly dropping out some nodes during training, effectively averaging over multiple different models.

<u>Use of batch normalisation</u>: Batch normalisation is a technique that helps to improve the stability and speed of training deep neural networks.

Popular Industrial Tools used for Deep Learning

TensorFlow

It is an open-source machine learning framework for all developers. It is used for implementing machine learning and deep learning applications.

Google team created TensorFlow to develop and research on fascinating ideas. TensorFlow is designed in python programming language hence it is considered as an easy to understand framework.

TensorFlow includes a variety of machine learning and deep learning algorithms. TensorFlow can train and run deep neural networks for handwritten digit classification, image recognition, word embedding and creation of various sequence models.

TensorFlow is most famous symbolic maths library used for creating neural networks and deep learning models.

Keras

It is a high-level deep learning API developed by Google for implementing neural networks. It is written in python and is used to make implementation of neural network easy. It also supports multiple backend neural network computation.

Keras runs on top of open-source machine libraries like TensorFlow, Theano or Cognitive toolkit. Keras is based on minimal structure that provides the clean and easy way to create deep learning models based on TensorFlow and Theano.

Keras is designed to quickly define deep learning models. Keras is an optimal choice for deep learning applications.

PyTorch

It is an open-source machine learning library for python and is completely based on Torch. It is primarily used for applications such as natural language processing.

PyTorch is developed by Facebook's artificial intelligence research group along with Ubers 'Pyro' software for the concept of inbuilt probabilistic programming.

PyTorch redesigns and implements Torch in Python while sharing the same core C libraries for the backend code.

PyTorch developers tuned this backend code to run python efficiently. They also kept the GPU based hardware acceleration as well as the extensibility feature that made Lua-based torch.

Shogun

It is an open-source machine learning software library built in C++. It offers a wide range of efficient and unified machine learning algorithms.

The heart of Shogun lies in kernel machines such as support vector machines for regression and classification problems.

Shogun offers a full implementation of Hidden Markov models. Its core is written in C++, and it offers interface for MATLAB, Octave, Python, R, Java, Lua, Ruby and C#.

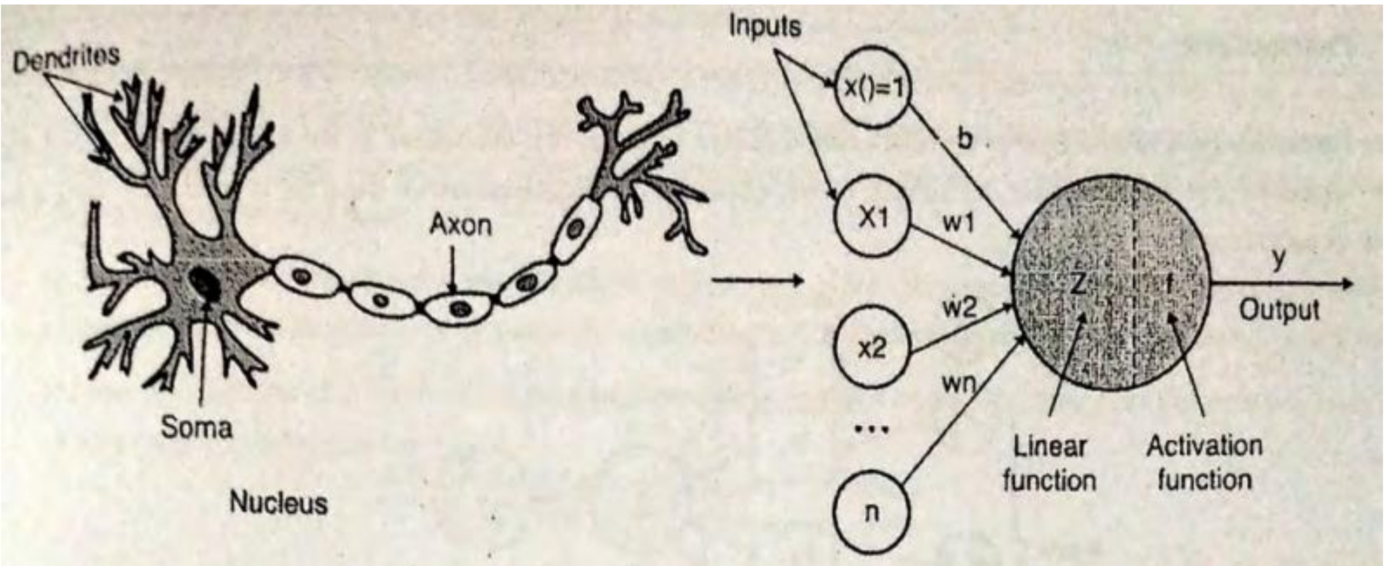
The Shogun ML toolkit encourages this expedition through its features (accessible, open source and good) and pivot on ML education and development. Shogun is one of the largest and oldest open source MLL platform.

Biological neuron

The nervous system is responsible for receiving sensory information from the environment, processing it, and sending signals to the appropriate organs and muscle to respond.

The fundamental unit of nervous system are biological neurons which collaborate to process and send information.

They play a critical role in sensory perception, motor control, and various cognitive processes, including learning and memory.



The basic structure of neuron includes the following:

Cell body (Soma): The cell body contains the nucleus which houses the genetic material of the cell and other organelles that are necessary for the function of the cell. Signals are sent to neighbouring neurons and muscles by long, thin Axon, which extends from the cell body.

Dendrites: Dendrites are the branch structure that extend from the cell body and receive signal from other neurons.

Axon: The Axon is long slender fibre that extends from the cell body and carries signals away from the cell body to other neurons, muscles, or glands.

Synapse: A synapse is a junction between the Axon of one neuron and the dendrites or cell body of another neuron where the two neurons communicate through the release of chemicals called neurotransmitter.

Perceptron

Perceptron is a simple type of artificial neural network. It was one of the first algorithm to be developed for solving binary classification problem where the goal is to classify data into one of two classes.

A Perceptron is a type of artificial neuron that is used in neural network to classify input data into one of two categories, based on a set of weights and threshold.

The perceptron takes input from several sources and applies a weight to each input. This determines how important each input is in making the decision.

The perception then calculates the weighted sum of these inputs and compares this sum to a threshold value.

If the sum is greater than the threshold, the perceptron output is 1, indicating that the input belongs to one category, and if the sum is less than the threshold, the perception output is 0, indicating that the input belongs to another category.

Single layer feedforward network

This type of network comprises of two layers, the input layer, and the output layer.

The input layer neurons receive the input signals, and the output layer neurons receive the output signals.

The synaptic links carrying the weights connect every input neuron to the output neuron but not in the reverse.

Such a network is said to be feedforward in type or acyclic in nature. Since the output layer alone performs computation, hence it is called as single layer feedforward network.

Multilayer feedforward network

A sort of artificial neural network called multilayer feedforward network, commonly referred to as multilayer perception is made up of numerous interconnected layers of artificial neurons.

Each layer in an MLP consist of a set of artificial neurons which are connected to the neurons in the previous subsequent layer.

The input layer provides the input in the output layer generates the final prediction.

The layer between the input and output layer are called as hidden layers, and they help the network learn more complex representation of the data.

Training Neural Networks

Training a neural network involves adjusting the values of its weights and biases so that the network can accurately predict the output for given input.

This process is usually performed using supervised learning where the network is presented with labelled example and the weight, and biases are updated based on the prediction error.

Back propagation and forward propagation are two important algorithms used in the training of neural networks.

Back Propagation

The back propagation algorithm is a supervised learning algorithm used to train artificial neural network with multiple layers, also known as multiple layer perceptron (MLPs).

Back propagation is the process of updating the weights and biases of a neural network during training. It is used to adjust the weight and bias so that the network can make accurate predictions of the training examples.

Back propagation utilises the gradient of the loss function with regards to the networks weight and biases to calculate the updates.

The architecture of a back propagation network typically consists of an input layer, one or more hidden layers and an output layer. The number of hidden layers and neurons in each layer can vary depending on the complexity of the task.

Forward Propagation

The Forward propagation algorithm also known as feedforward neural network is a type of artificial neural network that is commonly used for classification and regression tasks.

Forward propagation is the process of using the weights and biases of neural network to make predictions for a given input. The architecture of a feedforward neural network consists of an input layer, one or more hidden layers and an output layer.

Each layer contains multiple neurons that perform a weighted sum of their inputs and apply an activation function to the result. The activations are then passed through an activation function, which produces the final output of the network.

Activation functions

Activation functions are mathematical function used in artificial neural network to introduce non-linearity into the output of each neuron.

They are applied to the weighted sum of inputs to produce the neurons activation, which is used as input to the next layer in the network.

There are several types of activation functions used in neural network, including:

Sigmoid: The sigmoid function maps any input to a value between 0 and 1, making it useful for binary classification problem.

The sigmoid function is a smooth, S-shaped curve that allows it to model complex relationship between the inputs and outputs.

Rectified linear unit (ReLU): the ReLU activation function maps, any input below 0 to 0 and any input above 0 to the same value.

ReLU is computationally efficient and has been found to work well in many practical applications.

The choice of activation function depends on the specific problem being solved and the desired properties of the network.

Loss Functions

Loss functions, also known as cost functions or objective functions are used in machine learning to measure the difference between the predicted outputs of a model and the true outputs.

The goal of training a machine learning model is to minimise the value of the loss function.

There are several different types of loss functions, including mean squared error (MSE), mean absolute error (MAE), categorical cross entropy, binary cross entropy, and hinge loss.

The choice of loss function depends on the specific problem being solved and the type of model being used.

Mean square error (MSE): The MSE loss function is defined as the average of the square difference between the predicted values and the true values. The equation of MSE is given by:

$$J(\theta) = 1/n * \Sigma(i=1 \text{ to } n)\ (y\_i - \hat{y}\_i)\text{^}2$$

Where y_i is the true output for the i^{th} training example $y^{\hat{}}_i$ is the predicted output and n is the number of training examples.

Mean absolute error (MAE): The MAE loss function is defined as the average of the absolute difference between the predicted values and the true values. The equation of MAE is given by:

$$J(\theta) = 1/n * \Sigma(i=1 \text{ to } n)\ |y\_i - \hat{y}\_i|$$

Hyperparameters

Hyper parameters are parameters in a machine learning model that are set prior to training and control the the overall behaviour of the model. They are different from model parameters, which are learnt from data during the training process. Some common hyper parameters in neural networks included:

Number of layers: The number of layers in a neural network can have a large impact on its performance. A deeper network with more layers can learn more complex representation of data but can also be more difficult to train due to the risk of overfitting.

Number of units per layer: The number of units (neurons) in each layer can also impact the performance of the network. Large number of units can allow the network to learn more complex representation of the data but can also lead to overfitting.

Learning rate: The learning rate controls the step size used to update the model parameters during training. A smaller learning rate will cause the model to converse more slowly but will also reduce the risk of overshooting the optimal solution. A larger learning rate will cause the model to converse more quickly but can cause the model to miss the optimal solution.

Mini batch size: The mini batch size is the number of training example used in each iteration of stochastic gradient decent (SGD). A larger mini batch size can speed up training but can also increase the risk of overfitting. A smaller mini size can slow down training but can also help prevent overfitting.

Regularisation strength: Regularisation is a technique used to reduce overfitting in a model. Common forms of regularisation in neural network include L1 regularisation, L2 regularisation and dropout. The regularisation strength controls the amount of regularisation applied to the model.

Vanishing and Exploding Gradients

The vanishing and exploding gradient decent are two common problems in deep learning, particularly in deep feedforward neural networks (DFNNs).

The vanishing gradient problem occurs when the gradient becomes very small during back propagation, making it difficult for the optimisation algorithm to update the weight effectively.

This result in slow convergence or poor performance. The vanishing gradient problem is often encountered in deep networks because the gradients are multiplied multiple times during back propagation, making them smaller and smaller as they move from the output layer to the input layer.

The exploding gradient problem occurs when gradient become very large during back propagation, causing the optimisation algorithm to overshoot and produce large, unstable updates to the weights.

This results in unstable convergence or divergence. The exploding gradient problem is often encountered when the activation function used in the network have high values leading to large gradient during back propagation.

Sentiment analysis

Sentiment analysis is the process of determining the sentiment expressed in a piece of text, such as review, tweet, news article, etc.

The goal of sentiment analysis is to categorise the text as positive, negative, or neutral in terms of the sentiment expressed.

Sentiment analysis is a common task in natural language processing (NLP) and is used in wide range of applications, including customer feedback analysis, brand monitoring and opinion mining.

There are several approaches to sentiment analysis including rule-based method, which use a set of predefined rules to categorise the text and machine learning based method, which train models on large, annotated datasets to make prediction about the sentiment of new texts.

Sentiment analysis can be challenging task due to the subjectivity of language and the nuances of sentiment expression.

The accuracy of sentiment analysis model can be improved through the use of large annotated datasets, transfer learning, and the combination of multiple models.