NORTH SOUTH UNIVERSITY

Department of Electrical and Computer Engineering

Assignment 01

Instructor: Dr. Mohammad Abdul Qayum

Course: CSE 445

Section: 4

Submitted By:

-Kazi Abdullah Al Hasnaine

-2211688 642

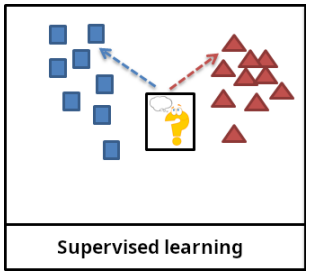
Online Copy

Google doc's : CSE 445 Assignment 1

Problem 1:

Part A: Fundamentals of Learning (1–10)

1. Supervised Learning

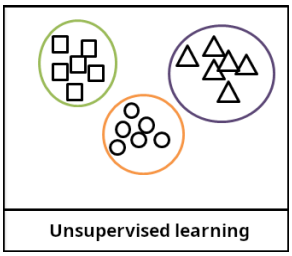


Definition:

A type of machine learning where the model is trained with labeled data. Each input has a matching correct output.

Elaboration:
The algorithm learns how to connect input variables (features) with the output (target) by reducing prediction errors. It is mainly used for classification, such as spam detection, and regression, like price prediction.

2. Unsupervised Learning



Definition:
A type of learning where the model is trained on unlabeled data. This means there are no predefined outputs.

Elaboration:
The algorithm looks for hidden structures, relationships, or patterns in the data. Common methods include clustering, like K-Means, which groups customers based on their behavior, and dimensionality reduction, such as PCA.

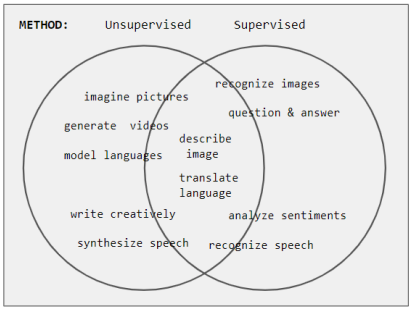
3. Semi-Supervised Learning

Definition:

A hybrid learning approach uses a little labeled data and a lot of unlabeled data.

Elaboration:

Since labeling data is costly, this method helps improve accuracy by using the few labeled samples to guide learning from the unlabeled ones. It’s commonly used in areas like image recognition and speech processing, where labeled data is hard to find.



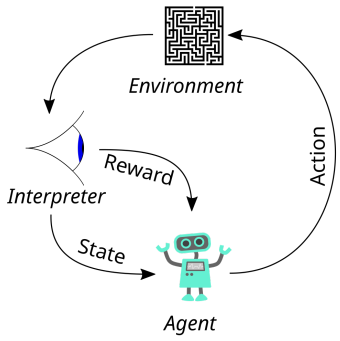
4. Reinforcement Learning

Definition:

A kind of learning involves an agent working with an environment to reach a goal. The agent learns from rewards or penalties.

Elaboration:

The model learns a policy or strategy that aims to maximize cumulative reward over time. It is commonly used in game AI, robotics, and autonomous systems. For example, it can train an AI to play chess or drive a car.



5. Feature / Attribute

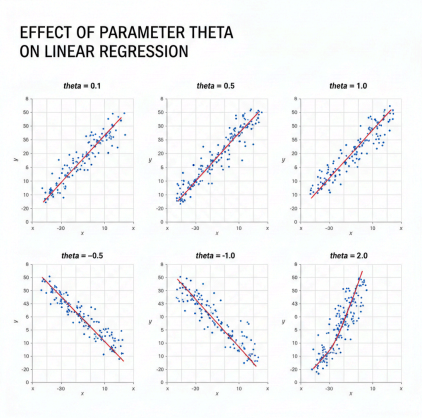
Definition:
A property or characteristic of the data that can be measured and used as input for a machine learning model.

Elaboration:
Features are the independent variables that describe each case in the dataset. For example, in a housing dataset, features might include area, the number of rooms, and location. Choosing the right features can improve model performance.

6. Parameter

Definition:
A configuration variable inside the model that comes from the training data.

Elaboration:
Parameters determine how the model behaves and makes predictions. For example, they include the weights and biases in a neural network or the coefficients in a linear regression model.



7. Training Data

Definition:
The part of the dataset used to train the model and change its parameters.

Elaboration:
It shows the model the basic patterns and connections between input and output. The amount and quality of training data have a big impact on the precision of the final model.

8. Validation Set

Definition:
A portion of the dataset was used to adjust the model's hyperparameters and avoid overfitting.

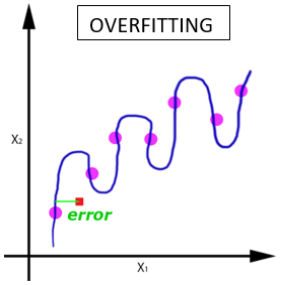
Elaboration:
The model does not learn from this data. Instead, it is tested during training to adjust settings, such as learning rate or number of layers, to achieve the best generalization.

9. Test Set

Definition:
A separate part of the dataset assesses the model's performance once training and validation are finished.

Elaboration:
It gives an unbiased estimate of how well the model will perform on unseen data. This is the final check before deployment.

10. Overfitting



Definition:

A modeling error happens when the algorithm learns the noise or details in the training data too well. This reduces its ability to apply what it learned to new data.

Elaboration:

An overfitted model does well on training data but struggles with test data. You can reduce overfitting using methods like regularization, dropout, or cross-validation.

Part B: Model and Optimization Concepts (11–20)

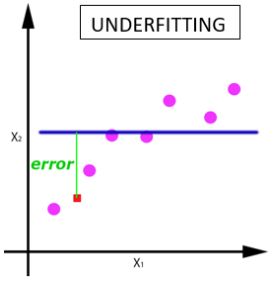
11. Underfitting

Definition:

A situation where the model is too simple to capture the patterns in the training data.

Elaboration:

The model performs poorly on both training and test data. The causes include too few features, a model that is too simple, or not enough training. Solutions involve increasing the model's complexity, adding features, or training for a longer period.



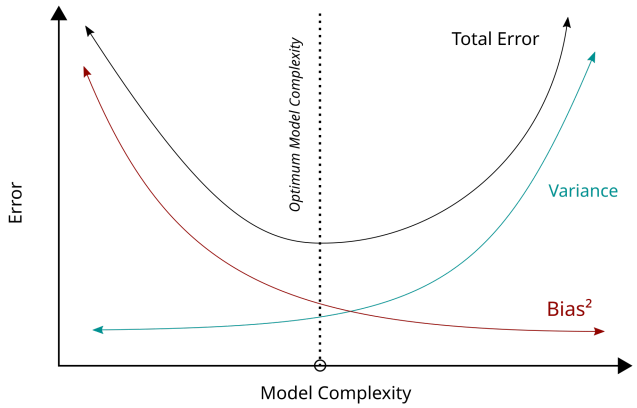
12. Bias–Variance Tradeoff

Definition:

The balance between a model's bias, which is the error caused by oversimplification, and its variance, which is the error from being sensitive to training data.

Elaboration:

High bias leads to underfitting. High variance results in overfitting. The goal is to find a model with both low bias and low variance so it can generalize well to new data.



13. Loss Function

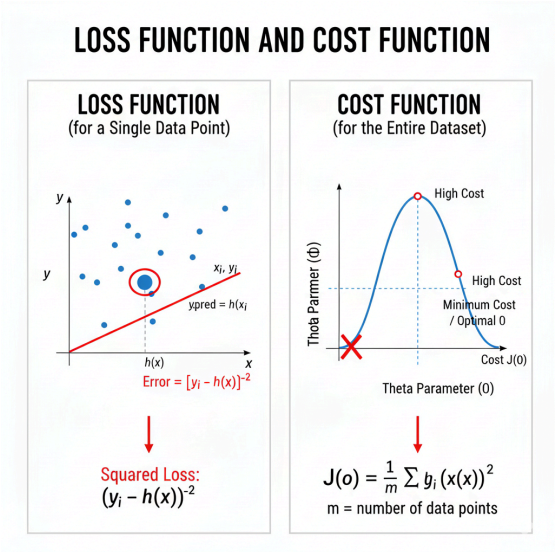
Definition:
A mathematical function measures the error between the predicted output and the actual output for one instance.

Elaboration:
Common examples include Mean Squared Error (MSE) for regression and Cross-Entropy Loss for classification. The model reduces this during training.

14. Cost Function

Definition:
The average loss over the entire dataset.

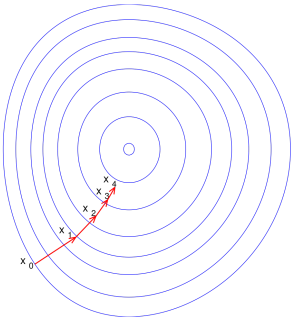
Elaboration:
While a loss function applies to each instance, the cost function sums it up across all training examples. The optimization algorithm, such as gradient descent, reduces the cost function to improve model performance.



15. Gradient Descent

Definition:
An optimization algorithm minimizes the cost function by updating model parameters in the direction of the negative gradient.

Elaboration:
It repeatedly changes weights and biases to minimize error. Variants include Batch Gradient Descent, Stochastic Gradient Descent (SGD), and Mini-batch Gradient Descent.

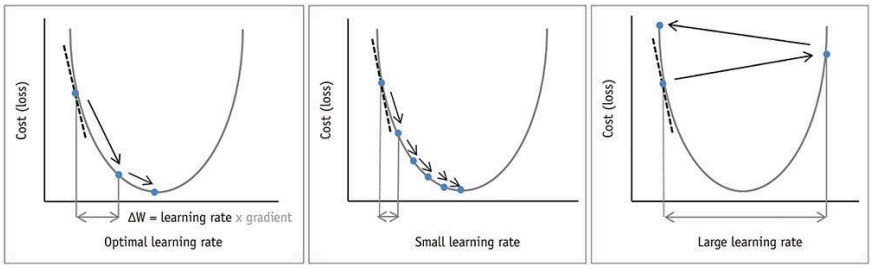


16. Learning Rate

Definition:
A hyperparameter that sets the size of the steps the optimizer takes during gradient descent.

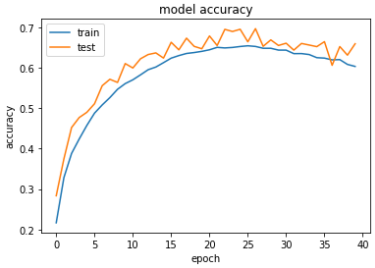
Elaboration:

Too large may overshoot the minimum. Too small leads to slow convergence. Choosing the right learning rate is important for effective training.



17. Epoch

Definition:
One full run of the training dataset through the model during training.



Elaboration:
Multiple epochs are often needed to make sure the model learns the patterns well. However, training for too many epochs can cause overfitting.

18. Cross-validation & Batch Size

Cross-validation

Definition:

A method to evaluate how well a model generalizes by dividing the dataset into several training and validation groups.

Elaboration:

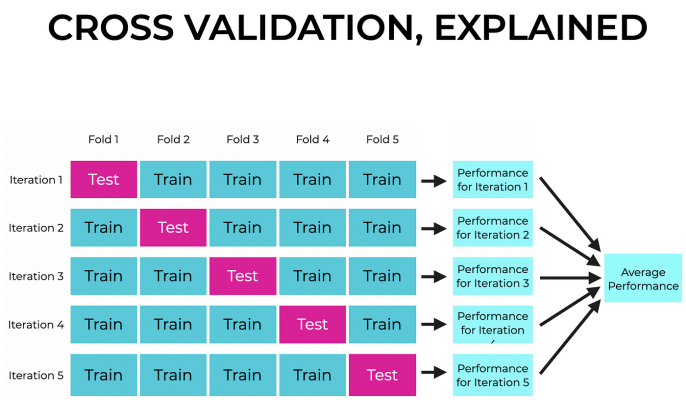
k-Fold cross-validation is common. The data is split into k parts. The model is trained on k-1 parts and validated on the remaining part. This process is repeated k times.

Batch Size Definition:

The number of training samples used in one iteration of gradient descent. Each fold is a batch in the graph.

Elaboration:

Small batch leads to a noisy gradient, which may result in faster generalization. Large batch produces a smoother gradient but requires more memory.



19. Weight Initialization

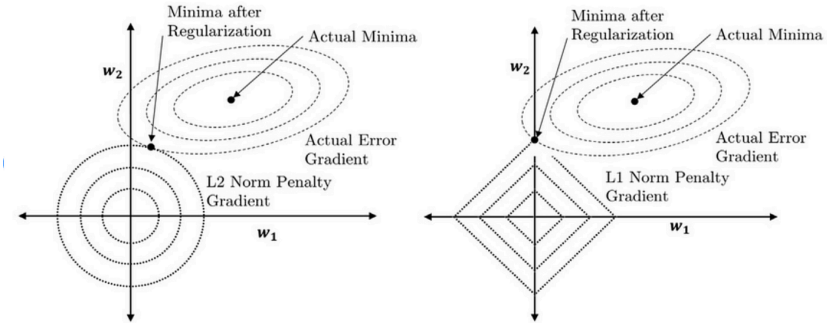
Definition:
The process of setting initial values for model parameters, or weights, happens before training.

Elaboration:
Proper initialization helps prevent issues like vanishing or exploding gradients. Common methods are Xavier or Glorot initialization and He initialization.

20. Regularization

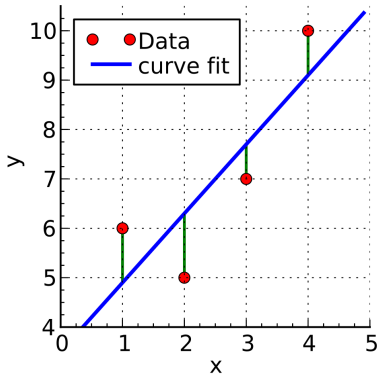
Definition:
A method to lessen overfitting by adding a penalty term to the cost function.

Elaboration:
Common methods:
L1 (Lasso) encourages sparsity.
L2 (Ridge) discourages large weights.
Dropout randomly disables neurons during training.



Part C: Core Algorithms & Techniques (21–30)

21. Linear Regression



Definition:

A supervised learning algorithm predicts a continuous numerical output based on one or more input features.

Elaboration:

It assumes a linear relationship between input(s) and output:

y=w1x1+w2x2+...+by = w_1x_1 + w_2x_2 + ... + by=w1x1+w2x2+...+b

It reduces the Mean Squared Error (MSE) between predicted and actual values. It is used in price prediction, sales forecasting, and other applications.

22. Logistic Regression

Definition:

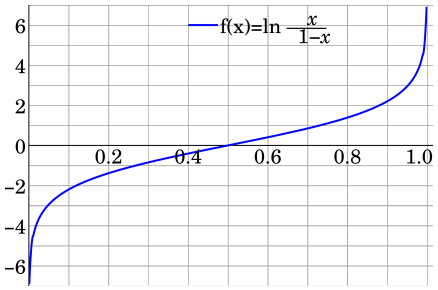
A supervised learning algorithm used for binary or multiclass classification.

Elaboration:

It predicts the probability of a class using the sigmoid function:

P(y=1)=11+e−zP(y=1) = \frac{1}{1 + e^{-z}}P(y=1)=1+e−z1

Despite the name, it's not for regression. It is often used in spam detection, disease diagnosis, and more.



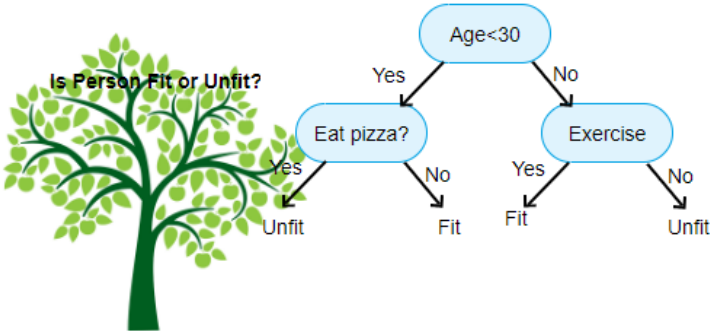
23. Decision Tree

Definition:
A tree-like model is used for classification and regression. It separates data into subsets based on feature values.

Elaboration:

- Nodes represent features.
- Branches represent decision rules.
- Leaves represent outcomes.

Advantages: easy to understand; handles both numerical and categorical data.
Disadvantages: liable to overfitting without pruning.

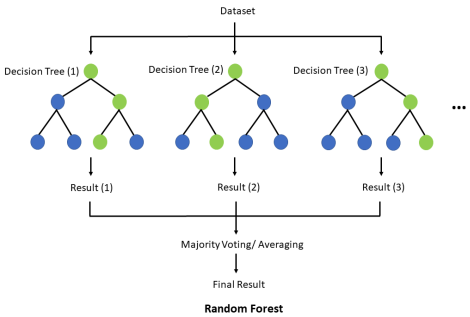


24. Random Forest

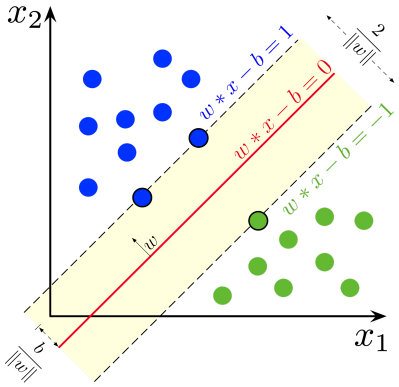
Definition:
An ensemble learning method builds several decision trees and combines their outputs to improve predictions.

Elaboration:

For classification, use the majority vote of trees. For regression, take the average of predictions. This method reduces overfitting when compared to a single decision tree and improves accuracy. It is useful for structured or tabular data.



25. Support Vector Machine (SVM)



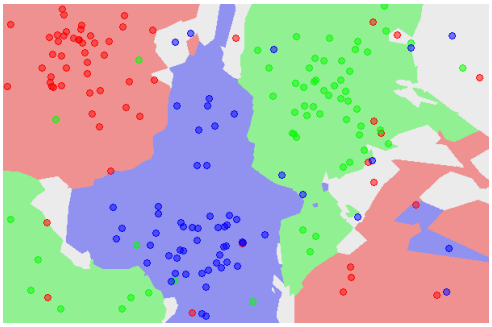
Definition:
A supervised algorithm finds the best hyperplane that separates data points from different classes.

Elaboration:

- Works well for both linear and non-linear data using kernel trick.
- Maximizes the margin between classes.

Used in text classification, image recognition, and small to medium sized datasets.

26. K-Nearest Neighbors (KNN)



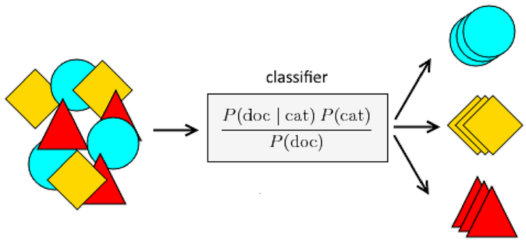
Definition:
A simple algorithm uses the instances of data points. Each point is classified according to the majority class of its k nearest neighbors.

Elaboration:

- No explicit training, lazy learning.
- Distance metrics like Euclidean or Manhattan are used.
- Sensitive to feature scaling and outliers.

Used for recommendation systems, pattern recognition, and small datasets.

27. Naïve Bayes



Definition:
A supervised learning algorithm based on Bayes' theorem assumes that features are independent.

Elaboration:

Predicts the likelihood of a class based on input features:

- P(C|X)=P(X|C)P(C)P(X)P(C|X) = \frac{P(X|C) P(C)}{P(X)}P(C|X)=P(X)P(X|C)P(C)

Despite the simple assumption, it works well in practice. This is especially true for text classification, spam detection, and sentiment analysis.

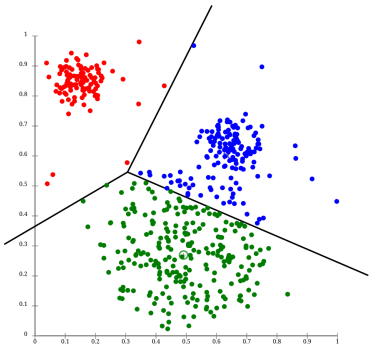
28. Principal Component Analysis (PCA)

Definition:
An unsupervised method for reducing dimensions changes data into a group of orthogonal components. These components capture the most variance.

Elaboration:

- Reduces the number of features while keeping most information.
- Aids in visualization, speeds up learning, and cuts down on noise.
- The first principal component shows the direction of maximum variance, the second shows the next orthogonal direction, and so on.

29. Clustering



Definition:
An unsupervised learning method groups similar data points into clusters using similarity metrics.

Elaboration:

- Does not need labeled data.
- Used for customer segmentation, spotting anomalies, and image segmentation.

30. Dimensionality Reduction

Definition:
The process of lowering the number of input features while keeping important information in the dataset.

Elaboration:
- Reduces computational cost, noise, and overfitting.
- Techniques include PCA, t-SNE, and LDA.
- Useful in high-dimensional datasets like image or genomic data.

Part D: Neural Network Foundations (31–40)

31. Perceptron

Definition:
A perceptron is the simplest part of a neural network. It takes its inspiration from how a single neuron functions in the human brain.

Elaboration:
It takes multiple input values. It multiplies these values by their weights and sums the results. Then, it uses an activation function to produce an output. If the weighted sum exceeds a certain threshold, the neuron "fires" and outputs 1; if not, it outputs 0. Perceptrons are the basis for more complex neural network architectures.

32. Activation Function

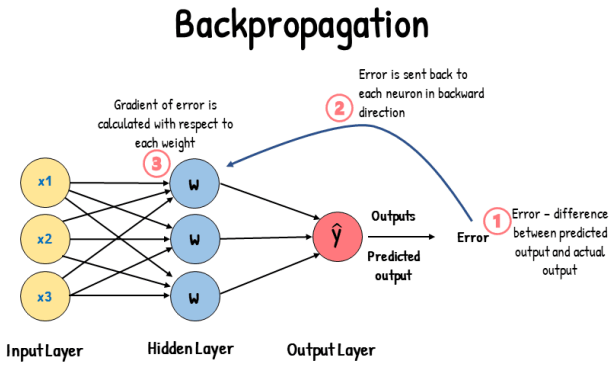
Definition:
An activation function decides if a neuron should be activated based on its input. Different types of activation functions exist.

Elaboration:
It introduces non-linearity into the model, which helps neural networks learn complex relationships. Common activation functions are Sigmoid, Tanh, and ReLU (Rectified Linear Unit). Without activation functions, neural networks would act like basic linear models and would not capture non-linear patterns.

33. Backpropagation

Definition:
Backpropagation is a learning method that trains neural networks by changing the weights in response to the error.

Elaboration:
It works by calculating the difference between predicted and actual outputs, which is the error. Then, it sends this error backward through the network using the chain rule of calculus. This process updates weights to reduce the loss function, helping the model get better over time.



34. Convolutional Neural Network (CNN)

Definition:
A CNN is a type of deep learning model mainly used for processing image data.

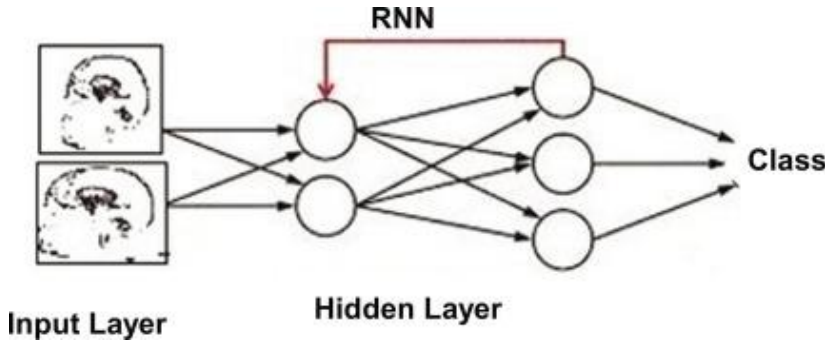
Elaboration:

It uses convolutional layers that automatically find spatial features like edges, shapes, and textures. Pooling layers lower dimensionality, and fully connected layers take care of classification. CNNs are commonly used in areas such as image recognition, object detection, and facial recognition.

35. Recurrent Neural Network (RNN)

Definition:
An RNN is a type of neural network designed for sequential or time-series data.

Elaboration:
Unlike standard neural networks, RNNs have loops that let information last between time steps. This makes them helpful for tasks like speech recognition, language modeling, and stock prediction. However, traditional RNNs face issues like vanishing gradients when dealing with long sequences.

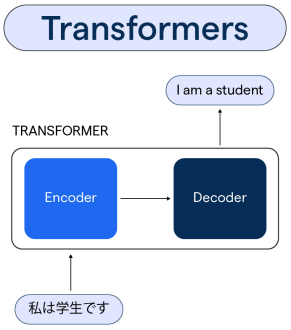


36. Long Short-Term Memory (LSTM)

Definition:
LSTM is a better type of RNN that can remember information for a long time.

Elaboration:
It includes special structures called gates, including input, forget, and output gates. These gates control what information to keep or discard. This setup helps LSTM models deal with the vanishing gradient problem. It also allows them to perform better on long-term dependencies in data, like text translation and speech generation.

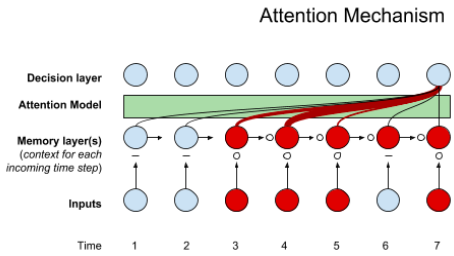
37. Transformer



Definition:
A Transformer is a deep learning model that uses the attention mechanism rather than recurrence.

Elaboration:
It processes whole sequences at the same time, which makes it faster and more efficient than RNNs. The Transformer has an encoder-decoder structure that uses self-attention to assess the importance of each part of the input. It is the basis for models like BERT and GPT used in natural language processing.

38. Attention Mechanism



Definition:
The attention mechanism lets a model focus on the most important parts of the input while it produces an output.

Elaboration:
Instead of treating all inputs the same, attention gives different weights based on importance. For instance, in language translation, it helps the model determine which words in the source sentence are most relevant to the word being generated. It boosts both accuracy and understanding in sequence models.

39. Dropout

Definition:
Dropout is a regularization technique used to prevent overfitting in neural networks.

Elaboration:
During training, dropout randomly turns off a fraction of neurons in a layer during each iteration. This encourages the network to learn stronger features instead of depending too much on certain neurons. During testing, all neurons are active, but their outputs are adjusted accordingly.

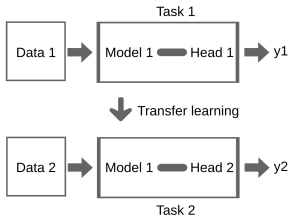
40. Gradient Vanishing Problem

Definition:
The gradient vanishing problem happens when gradients shrink too much during backpropagation. This stops effective learning.

Elaboration:
It often happens in deep networks or RNNs when using activation functions like Sigmoid or Tanh. As the gradients shrink through layers, earlier layers learn very slowly or not at all. Techniques such as using ReLU, batch normalization, or LSTM networks help reduce this problem.

Part E: Advanced Topics and Evaluation (41–50)

41. Transfer Learning



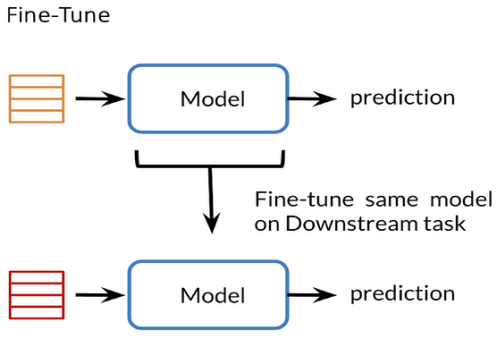
Definition:
Transfer learning is a technique that allows a model trained on one task to be used for another related task.

Elaboration:
Instead of starting from scratch, we can take what we learn from a pre-trained model, like those trained on ImageNet or large text datasets. This approach saves time, uses fewer resources, and often improves results in tasks like image classification or natural language processing.

42. Fine-Tuning

Definition:
Fine-tuning is the process of making small adjustments to a pre-trained model so it performs better on a specific new dataset.

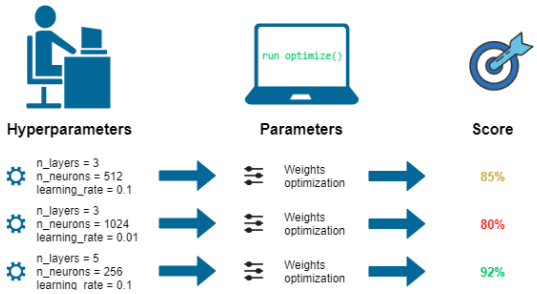
Elaboration:
This involves retraining only some layers of a pre-trained model with a small learning rate while keeping the earlier layers unchanged. This way, the model adapts its learned features to fit the new data better without losing previous knowledge. Fine-tuning is common in transfer learning setups.



43. Hyperparameter Optimization

Definition:
Hyperparameter optimization is the process of finding the best set of hyperparameters that improve a model’s performance.

Elaboration:
Hyperparameters are external settings like learning rate, batch size, or number of layers that the model does not learn during training. Techniques such as Grid Search, Random Search, and Bayesian Optimization test different combinations of hyperparameters to find the best accuracy and efficiency for the model.



44. Specificity

Definition:
Specificity measures how well a model identifies true negatives among all actual negatives.

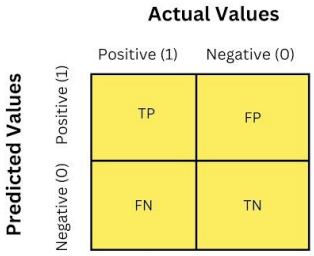
Elaboration:
It indicates the model's ability to correctly reject cases that don’t belong to a specific class. For example, in medical diagnosis, high specificity means the model correctly identifies healthy patients as not having the disease. It is calculated as:

Specificity = TN / (TN + FP)

where TN = True Negatives, FP = False Positives.

45. Confusion Matrix

Definition:
A confusion matrix is a table that summarizes how well a classification model performs.



Elaboration:
It shows the counts of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). By looking at these numbers, you can calculate metrics like accuracy, precision, recall, and F1 score. The confusion matrix helps you understand model performance beyond just accuracy.

46. Precision

Definition:
Precision measures how many of the predicted positive cases are actually positive.

Elaboration:
It focuses on the quality of positive predictions. High precision means the model makes fewer false positive errors. It is calculated as:
Precision = TP / (TP + FP)
Precision is especially important in situations where false positives are costly, like spam detection or medical diagnosis.

47. Recall

Definition:
Recall measures how many actual positive cases were correctly identified by the model.

Elaboration:
It shows how well the model can find all relevant instances. High recall means that it misses few positive cases. It is calculated as:
Recall = TP / (TP + FN)
Recall is important in areas like disease detection or fraud detection. Missing a positive case can have serious consequences.

48. F1 Score

Definition:
The F1 score is the harmonic mean of precision and recall, providing a balanced measure of both.

Elaboration:

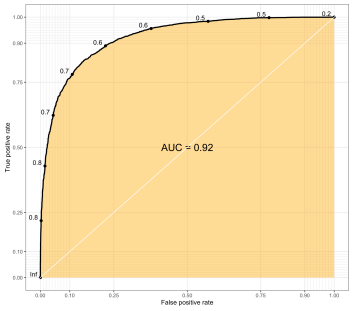
It combines precision and recall into one metric to evaluate model performance, especially with imbalanced data. The formula is:
$F1 = 2 \times (\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall})$
A high F1 score indicates that the model effectively finds all the positive cases while keeping false positives low.

49. ROC Curve

Definition:
The ROC (Receiver Operating Characteristic) curve is a graph that shows a model's performance across different classification thresholds.

Elaboration:
It shows the True Positive Rate (Recall) compared to the False Positive Rate (1 - Specificity). Each point represents a possible threshold for making positive or negative predictions. The ROC curve helps illustrate how sensitivity and specificity balance out for different thresholds.

50. AUC (Area Under Curve)



Definition:
AUC stands for Area Under the ROC Curve and measures how well a model can distinguish between classes.

Elaboration:
An AUC value ranges from 0 to 1. A value of 1 means perfect classification, while 0.5 indicates random guessing. Higher AUC values show better performance. AUC is commonly used to compare classification models because it summarizes the ROC curve in one number.

Problem 2:

Assignment-01.ipynb submitted and
Github : [Assignment 1]

Problem 3:

Impact of Vibe Coding on Software Quality

With vibe coding (code written with AI tools like ChatGPT, Grok, and Gemini) becoming more prevalent, our project is focused on comparing it to traditional human-written code based on code quality. We are planning to use the QScored dataset for this task. We will be training our model to detect poor code quality based on long methods/functions. After that, we'll use the model to compare human-written code and AI-generated code to observe who produces cleaner code.