Graduate Certificate in Machine Learning in Polymer Science and Engineering

Deep Learning and Neural Networks

Deep Learning:
Deep learning is a subset of machine learning that utilizes artificial neural networks with multiple layers to model and extract high-level features from data. It involves training neural networks on large datasets to learn complex patterns and representations. Deep learning algorithms have shown great success in various applications such as image and speech recognition, natural language processing, and autonomous driving.

Neural Networks:
Neural networks are a class of machine learning algorithms inspired by the structure and function of the human brain. They consist of interconnected nodes, or neurons, organized into layers. Each neuron receives input, processes it through an activation function, and produces an output. Neural networks can learn to perform tasks by adjusting the strength of connections between neurons during training.

Activation Function:
An activation function is a mathematical function applied to the output of a neuron in a neural network. It introduces non-linearity to the network, allowing it to learn complex patterns in the data. Common activation functions include the sigmoid, tanh, ReLU (Rectified Linear Unit), and softmax functions.

Backpropagation:
Backpropagation is a key algorithm used to train neural networks. It involves calculating the gradient of the loss function with respect to the weights of the network and updating the weights in the opposite direction to minimize the loss. Backpropagation is an iterative process that adjusts the network parameters to improve its performance.

Convolutional Neural Network (CNN):
A Convolutional Neural Network (CNN) is a type of neural network designed for processing structured grid data, such as images. CNNs use convolutional layers to extract features from input data, pooling layers to reduce dimensionality, and fully connected layers for classification. CNNs have achieved state-of-the-art performance in image recognition tasks.

Recurrent Neural Network (RNN):
A Recurrent Neural Network (RNN) is a type of neural network that is well-suited for sequence data, such as time series or natural language. RNNs have feedback connections that allow them to maintain a memory of previous inputs, making them effective for tasks that require understanding context and long-term dependencies.

Long Short-Term Memory (LSTM):
Long Short-Term Memory (LSTM) is a type of recurrent neural network architecture that addresses the vanishing gradient problem in traditional RNNs. LSTMs have a more complex structure with gated cells that control the flow of information, allowing them to learn and remember long-term dependencies in

sequential data.

Autoencoder:
An autoencoder is a type of neural network that learns to encode input data into a lower-dimensional representation and then decode it back to the original input. Autoencoders are used for unsupervised learning, dimensionality reduction, and feature learning. They consist of an encoder that compresses the input and a decoder that reconstructs it.

Generative Adversarial Network (GAN):
A Generative Adversarial Network (GAN) is a type of neural network architecture that consists of two networks, a generator and a discriminator, trained simultaneously. The generator learns to generate realistic samples from random noise, while the discriminator learns to distinguish between real and generated samples. GANs are used for generating synthetic data and creating realistic images.

Overfitting:
Overfitting occurs when a machine learning model performs well on the training data but poorly on unseen data. It happens when the model learns noise or irrelevant patterns in the training data, rather than the underlying structure. Overfitting can be mitigated by using regularization techniques, cross-validation, and early stopping.

Underfitting:
Underfitting occurs when a machine learning model is too simple to capture the underlying patterns in the data. It results in poor performance on both the training and test data. Underfitting can be addressed by using more complex models, increasing the model capacity, or collecting more data.

Dropout:
Dropout is a regularization technique used to prevent overfitting in neural networks. During training, a fraction of neurons in the network are randomly set to zero, effectively dropping them out of the network. This forces the network to learn more robust and generalizable features.

Batch Normalization:
Batch normalization is a technique used to improve the training of deep neural networks by normalizing the input to each layer. It helps stabilize and speed up the training process by reducing internal covariate shift. Batch normalization is applied after the activation function in each layer of the network.

Hyperparameter:
Hyperparameters are parameters that are set before training a machine learning model and control the learning process. Examples of hyperparameters include learning rate, batch size, number of layers, and activation functions. Tuning hyperparameters is essential for optimizing the performance of a model.

Loss Function:
A loss function is a measure of how well a machine learning model's predictions match the actual target values. It quantifies the error between the predicted and true values and is used to update the model parameters during training. Common loss functions include mean squared error, cross-entropy, and hinge loss.

Gradient Descent:
Gradient descent is an optimization algorithm used to minimize the loss function of a machine learning model by adjusting its parameters. It calculates the gradient of the loss function with respect to the model parameters and updates the parameters in the opposite direction to the gradient. Gradient descent is an iterative process that converges to a local minimum.

Learning Rate:
The learning rate is a hyperparameter that controls the size of the steps taken during gradient descent optimization. It determines how quickly the model parameters are updated based on the gradient of the loss function. Choosing an appropriate learning rate is crucial for training neural networks effectively.

Regularization:
Regularization is a technique used to prevent overfitting in machine learning models by adding a penalty term to the loss function. Common regularization methods include L1 regularization (Lasso), L2 regularization (Ridge), and dropout. Regularization helps to reduce the complexity of the model and improve its generalization ability.

Feature Engineering:
Feature engineering is the process of selecting, transforming, and creating new features from raw data to improve the performance of machine learning models. It involves domain knowledge, data preprocessing, and feature selection techniques. Effective feature engineering can significantly impact the accuracy of a model.

Cross-Validation:
Cross-validation is a technique used to evaluate the performance of a machine learning model by splitting the data into multiple subsets. The model is trained on a portion of the data and tested on the remaining data, with the process repeated multiple times. Cross-validation helps assess the generalization ability of a model.

Transfer Learning:
Transfer learning is a machine learning technique where a model trained on one task is adapted to a new, related task. It allows the knowledge learned from one domain to be transferred to another domain, reducing the amount of data and training time required for the new task. Transfer learning is commonly used in deep learning for image classification.

Optimization Algorithm:
An optimization algorithm is a method used to adjust the parameters of a machine learning model to minimize the loss function. Common optimization algorithms include gradient descent, stochastic gradient descent, and Adam. These algorithms play a crucial role in training neural networks efficiently.

Kernel:
In machine learning, a kernel is a function that computes the similarity between pairs of data points in a higher-dimensional space. Kernels are used in support vector machines (SVMs) and kernel methods to transform non-linearly separable data into linearly separable data. Common kernels include linear,

polynomial, and radial basis function (RBF) kernels.

Dimensionality Reduction:
Dimensionality reduction is the process of reducing the number of features in a dataset while preserving important information. It helps simplify the data, reduce noise, and improve the performance of machine learning models. Common techniques for dimensionality reduction include principal component analysis (PCA) and t-distributed stochastic neighbor embedding (t-SNE).

Ensemble Learning:
Ensemble learning is a machine learning technique that combines multiple models to improve predictive performance. It involves training a set of diverse models and aggregating their predictions to make a final decision. Ensemble methods such as bagging, boosting, and stacking are commonly used to create robust and accurate models.

Reinforcement Learning:
Reinforcement learning is a type of machine learning where an agent learns to make decisions by interacting with an environment. The agent receives rewards or penalties for its actions, which guide it towards learning an optimal policy. Reinforcement learning is used in applications such as game playing, robotics, and autonomous driving.

Hyperparameter Tuning:
Hyperparameter tuning is the process of selecting the best hyperparameters for a machine learning model to optimize its performance. It involves searching through a predefined space of hyperparameters using techniques such as grid search, random search, and Bayesian optimization. Hyperparameter tuning is essential for achieving the best results with a model.

Unsupervised Learning:
Unsupervised learning is a type of machine learning where the model learns patterns and relationships in the data without explicit labels. It involves clustering, dimensionality reduction, and density estimation tasks. Unsupervised learning is used for exploratory data analysis, anomaly detection, and data visualization.

Supervised Learning:
Supervised learning is a type of machine learning where the model learns from labeled training data to make predictions on new, unseen data. It involves mapping input features to target outputs and is used for regression and classification tasks. Supervised learning algorithms include linear regression, support vector machines, and neural networks.

Deep Reinforcement Learning:
Deep reinforcement learning is a combination of deep learning and reinforcement learning techniques used to solve complex sequential decision-making problems. It involves training deep neural networks to learn policies that maximize cumulative rewards in an environment. Deep reinforcement learning has achieved remarkable success in games and robotics.

Clustering:
Clustering is a type of unsupervised learning where data points are grouped into clusters based on their

similarity. It aims to find natural groupings in the data without any prior knowledge of the labels. Common clustering algorithms include K-means, hierarchical clustering, and DBSCAN.

K-means:
K-means is a popular clustering algorithm that partitions data points into K clusters based on their distances to the cluster centroids. It aims to minimize the sum of squared distances within each cluster. K-means is an iterative algorithm that converges to a local optimum and is sensitive to the initial cluster centers.

Hierarchical Clustering:
Hierarchical clustering is a clustering algorithm that organizes data points into a tree-like hierarchy of clusters. It can be agglomerative, where each data point starts as a separate cluster and is sequentially merged, or divisive, where all data points start in one cluster and are recursively split. Hierarchical clustering does not require the number of clusters to be specified in advance.

DBSCAN:
DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a clustering algorithm that groups together data points based on their density. It identifies clusters as areas of high density separated by areas of low density. DBSCAN is robust to noise and can find clusters of arbitrary shapes and sizes.

Principal Component Analysis (PCA):
Principal Component Analysis (PCA) is a dimensionality reduction technique that transforms high-dimensional data into a lower-dimensional space while preserving the most important information. It identifies the directions, or principal components, of maximum variance in the data and projects the data onto these components. PCA is widely used for visualization and data compression.

t-Distributed Stochastic Neighbor Embedding (t-SNE):
t-Distributed Stochastic Neighbor Embedding (t-SNE) is a nonlinear dimensionality reduction technique used for visualizing high-dimensional data in a low-dimensional space. It aims to preserve the local structure of the data points in the original space. t-SNE is commonly used for visualizing clusters and patterns in complex datasets.

Support Vector Machine (SVM):
A Support Vector Machine (SVM) is a supervised learning algorithm used for classification and regression tasks. It works by finding the hyperplane that best separates the classes in the feature space. SVMs maximize the margin between the classes and are effective for high-dimensional data and non-linear problems using kernel functions.

Kernel Trick:
The kernel trick is a technique used in support vector machines and kernel methods to implicitly map data points into a higher-dimensional space. It allows SVMs to find nonlinear decision boundaries by defining a kernel function that computes the dot product in the transformed space. Common kernels include linear, polynomial, and radial basis function (RBF) kernels.

Bagging:
Bagging, short for bootstrap aggregating, is an ensemble learning technique that combines multiple

models trained on different subsets of the data. Each model is trained independently, and their predictions are aggregated to make a final decision. Bagging helps reduce variance and improve the stability of the model.

Boosting:
Boosting is an ensemble learning technique that builds a strong learner by sequentially training weak learners and giving more weight to misclassified instances. Each weak learner focuses on the mistakes of the previous learners, improving the overall performance. Boosting algorithms include AdaBoost, Gradient Boosting, and XGBoost.

Stacking:
Stacking, also known as stacked generalization, is an ensemble learning technique that combines multiple diverse models to improve predictive performance. In stacking, the predictions of base models are used as input features for a meta-model, which learns to make the final prediction. Stacking can capture complementary patterns from different models.

Recurrent Neural Network (RNN):
A Recurrent Neural Network (RNN) is a type of neural network that is well-suited for sequence data, such as time series or natural language. RNNs have feedback connections that allow them to maintain a memory of previous inputs, making them effective for tasks that require understanding context and long-term dependencies.

Long Short-Term Memory (LSTM):
Long Short-Term Memory (LSTM) is a type of recurrent neural network architecture that addresses the vanishing gradient problem in traditional RNNs. LSTMs have a more complex structure with gated cells that control the flow of information, allowing them to learn and remember long-term dependencies in sequential data.

Autoencoder:
An autoencoder is a type of neural network that learns to encode input data into a lower-dimensional representation and then decode it back to the original input. Autoencoders are used for unsupervised learning, dimensionality reduction, and feature learning. They consist of an encoder that compresses the input and a decoder that reconstructs it.

Generative Adversarial Network (GAN):
A Generative Adversarial Network (GAN) is a type of neural network architecture that consists of two networks, a generator and a discriminator, trained simultaneously. The generator learns to generate realistic samples from random noise, while the discriminator learns to distinguish between real and generated samples. GANs are used for generating synthetic data and creating realistic images.

Overfitting:
Overfitting occurs when a machine learning model performs well on the training data but poorly on unseen data. It happens when the model learns noise or irrelevant patterns in the training data, rather than the underlying structure. Overfitting can be mitigated by using regularization techniques, cross-validation, and

early stopping.

Underfitting:
Underfitting occurs when a machine learning model is too simple to capture the underlying patterns in the data. It results in poor performance on both the training and test data. Underfitting can be addressed by using more complex models, increasing the model capacity, or collecting more data.

Dropout:
Dropout is a regularization technique used to prevent overfitting in neural networks. During training, a fraction of neurons in the network are randomly set to zero, effectively dropping them out of the network. This forces the network to learn more robust and generalizable features.

Batch Normalization:
Batch normalization is a technique used to improve the training of deep neural networks by normalizing the input to each layer. It helps stabilize and speed up the training process by reducing internal covariate shift. Batch normalization is applied after the activation function in each layer of the network.

Hyperparameter:
Hyperparameters are parameters that are set before training a machine learning model and control the learning process. Examples of hyperparameters include learning rate, batch size, number of layers, and activation functions. Tuning hyperparameters is essential for optimizing the performance of a model.

Loss Function:
A loss function is a measure of how well a machine learning model's predictions match the actual target values. It quantifies the error between the predicted and true values and is used to update the model parameters during training. Common loss functions include mean squared error, cross-entropy, and hinge loss.

Gradient Descent:
Gradient descent is an optimization algorithm used to minimize the loss function of a machine learning model by adjusting its parameters. It calculates the gradient of the loss function with respect to the model parameters and updates the parameters in the opposite direction to the gradient. Gradient descent is an iterative process that converges to a local minimum.

Learning Rate:
The learning rate is a hyperparameter that controls the size of the steps taken during gradient descent optimization. It determines how quickly the model parameters are updated based on the gradient of the loss function. Choosing an appropriate learning rate is crucial for training neural networks effectively.

Regularization:
Regularization is a technique used to prevent overfitting in machine learning models by adding a penalty term to the loss function. Common regularization methods include L1 regularization (Lasso), L2 regularization (Ridge), and dropout. Regularization helps to reduce the complexity of the model and improve its generalization ability.

Feature Engineering:
Feature engineering is the process of selecting, transforming, and creating new features from raw data to improve the performance of machine learning models. It involves domain knowledge, data preprocessing, and feature selection techniques. Effective feature engineering can significantly impact the accuracy of a model.

Cross-Validation:
Cross-validation is a technique used to evaluate the performance of a machine learning model by splitting the data into multiple subsets. The model is trained on a portion of the data and tested on the remaining data, with the process repeated multiple times. Cross-validation helps assess the generalization ability of a model.

Transfer Learning:
Transfer learning is a machine learning technique where a model trained on one task is adapted to a new, related task. It allows the knowledge learned from one domain to be transferred to another domain, reducing the amount of data and training time required for the new task. Transfer learning is commonly used in deep learning for image classification.

Optimization Algorithm:
An optimization algorithm is a method used to adjust the parameters of a machine learning model to minimize the loss function. Common optimization algorithms include gradient descent, stochastic gradient descent, and Adam. These algorithms play a crucial role in training neural networks efficiently.

Kernel:
In machine learning, a kernel is a function that computes the similarity between pairs of data points in a higher-dimensional space. Kernels are used in support vector machines (S