**Lab Sheet – 6**

Question 2

|  |  |  |
| --- | --- | --- |
| **Supervised Learning** | **Self Supervised Learning** | **Semi-Supervised Learning** |
| Learn from labeled data where each training example has a known label. | Learns from data itself, with no labeled examples (Unlabeled data) | Uses a combination of labeled data and unlabeled data for training |
| algorithm is provided with a set of input data and the corresponding desired output for each example. It then uses this information to learn how to produce the correct output for new, unlabeled data. | algorithm has to find structure in the data in order to learn from it. | The computer is first given a set of training data (usually a set of labeled data) and is then trained to recognize patterns in that data. Once the computer is able to recognize patterns in the data, it is then given a set of unlabeled data and is asked to find patterns in that data. The computer can then use the patterns it found in the labeled data to help it find patterns in the unlabeled data. |

|  |  |
| --- | --- |
| **Transductive learning** | **Inductive learning** |
| Make predictions on specific, unseen data without explicitly learning a general model. | Generalize from training data to predict unseen data. |
| directly infers labels for the unseen examples without a general model | Build a model for future predictions |
| Generalization is limited because only applied for current test data | Generalization is high because applies for any new data |

Question 3

Increase epoch from 50 and 500 and below is the observation

* As the number of epochs increases, the model continues to adjust its parameters to minimize the loss function. At epoch 50, the loss is high (1.4324), meaning the model hasn't fully learned. By epoch 250, the loss has decreased significantly up to 0.0667. At epoch 500, the loss is further minimized to 0.0206.
* The training data reaches 100% at epoch 80 which means the model perfectly fits the training data. However, the validation accuracy is 73.53% at epoch 80. this indicates that the model is not generalized well.
* As training continues, validation accuracy increases and reaches a peak with a much lower loss (0.0393) at epoch 340 with an 82.35% validation accuracy. It means model learns to perform better on new data.

Observations without self-loops added to GCNConv() layers in the GCN() model

* Without self-loops, the validation accuracy decreased significantly, at higher epochs (47.09% at epoch 80 compared to 73.53% with self-loops , 61.76% at epoch 390 without self-loops but with self-loops its 82.35% at the same epoch).
* The training accuracy reach the similar level 100% in both with and without self-loops. But the model without self-loops took longer to achieve high accuracy (reaches 100% with self-loops at epoch 80 but without self-loops it reaches at epoch 370)

Increase the number of GCNConv() layers in the GCN() model upto 8 layers from original 3 layers – observations

* The highest validation accuracy achieved with 8 layers and without self-loops is 55.88%, which is reached at 330 epochs. It remains at 55.88% even at 390 epochs. This validation accuracy is lower compared to the original model with fewer layers and no self-loops
* The model with 8 layers and no self-loops reached 100% training accuracy at 240 epochs, which is faster compared to the scenario with fewer layers and no self-loops (reach 100% at epoch 370)

1. When out\_channels = 4 ,

The training accuracy reaches 100% at epoch 100 where validation accuracy is 35.29% and loss = 0.6574. the loss is decreasing gradually and at epoch 500 it's 0.076. the highest validation accuracy is 50% and it reaches epoch 140. but it again started to decrease from epoch 170 and at 500 epoch it is 44.12%

When out\_channels=8

The loss at epoch 500 is 0.0011 and validation accuracy is 52.94% training accuracy reaches to 100% at epoch the validation accuracy is 55.88% at epoch 300 and then decreases again. The training accuracy reaches 100 at epoch 80 at at epoch 50 the training accuracy is 0%. The loss is gradually decreasing while the number of epochs are increasing.

When out\_channels=16

At epoch 500 the training accuracy is 100% loss is 0.0007 and validation accuracy is 47.06%. The training accuracy reaches 100% at the epoch. Maximum valudation accuracy is 70.59% it reaches epoch 60 the training accuracy at epoch 60 is 75%. But in epoch 50 the validation accuracy is 14.71 % and the training accuracy is 25.00%. the validation accuracy decreased to 47.06% at epoch 70% and then again increase to 64.71% at epoch 80.

Best-performing value is 16 because it provides the highest validation accuracy and lowest loss.

1. When out\_channels=16, add skip connections between some of the GCNConv() layers. Here is the observation.

* Training Accuracy was achieved 100% early in the training process and remained stable.
* Validation Accuracy peaked at 55.88% around epochs 80 and 340, but started to decrease afterward, showing potential overfitting.
* Loss decreased consistently

Question 4

|  |  |  |  |
| --- | --- | --- | --- |
| **Message Passing GNN (MP-GNN)** | **Graph Convolution Network (GCN)** | **Graph Attention Network (GAT)** | **GraphSAGE** |
| General framework for defining GNNs. | Applies convolution operations to graph data. | Enhances GCN with attention mechanisms. | Handles large graphs by sampling fixed-size neighborhoods. |
| Message aggregation and node update. | Aggregates features from neighbors using graph convolution. | Assigns attention weights to neighbors' messages. | Aggregates information from sampled neighbors. |