Assignment: CS 768
Learning With Graphs

Richeek Das : 190260036

7th November 2021

Contents

Department of Computer Science and Engineering
Indian Institute of Technology Bombay

2021-2022

1 Implementation of the GNNFactory function

Implementation present in the shared `.ipynb` file.

Model description:

In the `def __init__()` function we set a torch seed to ensure reproducibility of the results we report in this work. We find out the `torch_geometric.nn` convolution layer which fits our `model_type`. After we are done with this, we stack a layer of convolution layers based on the `num_layers` parameter in our argument list.

In the `def forward()` function we apply one convolution layer, followed by a REctified Linear Unit, followed by a Dropout. We continue this till we exhaust all our convolution layers. We return a set of probabilities by performing a log_softmax of the outputs of the final convolution layer using: `F.log_softmax(x, dim=1)`.

2 Early Stopping

Find the code in the shared `.ipynb`.

We keep a track of the best epoch so far (based on the validation accuracy). We do early stopping if we find: `current_epoch` − `best_epoch` > 50. We also save the model weights of the best epochs as `.pkl` files. After training is done (either after total number of epochs or by early-stopping), we load the weights of the best epoch and report the test results.

3 Accuracy and Best Hyperparameters

(a): Hyperparameters for each of these cases to obtain the best performance:

model	dataset	num_layers	batch_size	hidden_dim	dropout	weight_decay	lr
GCN	cora	3	32	256	0.6	1e-2	0.01
	citeseer	3	32	32	0.6	1e-2	0.01
SAGE	cora	3	32	40	0.6	2e-2	0.009
	citeseer	3	32	32	0.6	2e-2	0.01
GAT	cora	3	32	256	0.6	1e-2	0.009
	citeseer	3	32	256	0.6	2e-2	0.009

Table 1: Hyperparameters used for obtaining best performance in each of the 6 cases. Note that `batch_size` doesn't matter in this problem setting.

(b): Training loss and validation accuracy across epochs in all 6 cases:

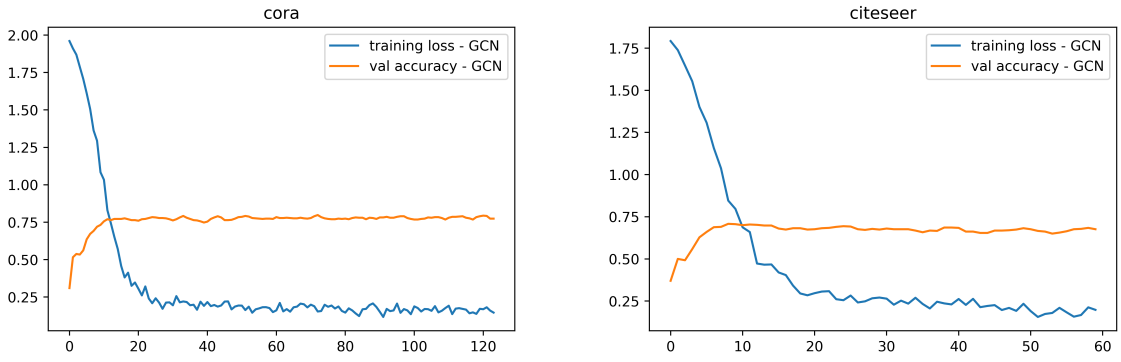


Figure 1: **GCN**. `xlabel: epochs`

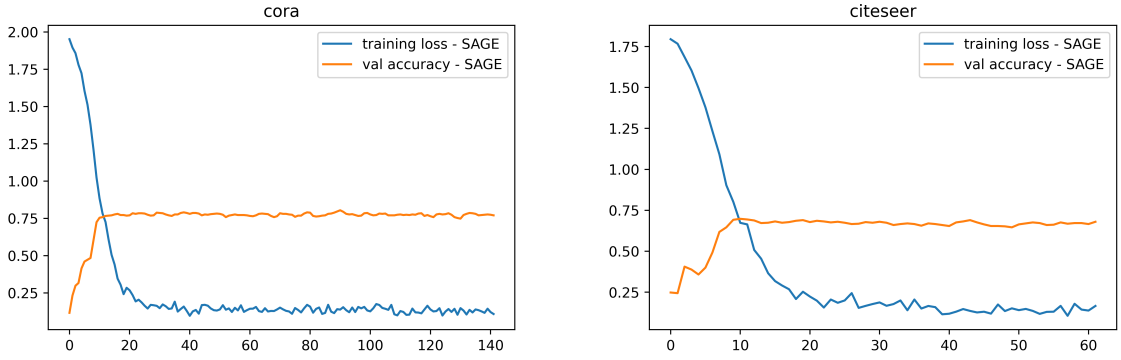


Figure 2: **SAGE**. `xlabel: epochs`

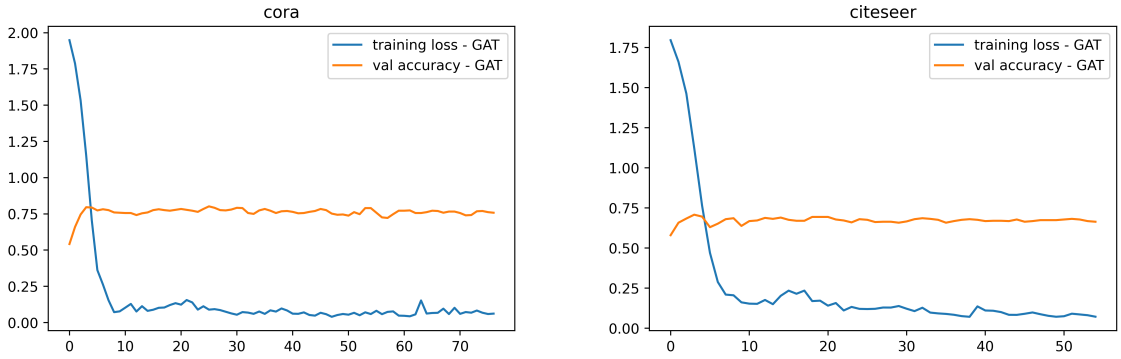


Figure 3: **GAT**. `xlabel: epochs`

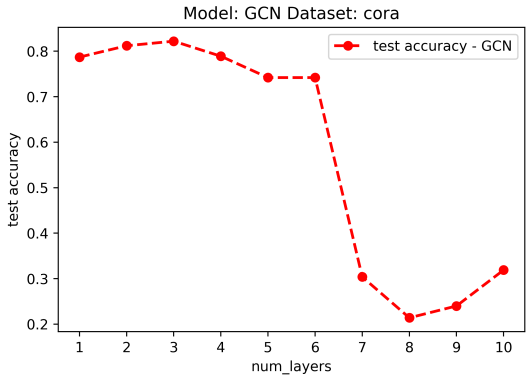
(c): Test accuracies with the above best hyperparameters:

Model \ Dataset	**cora**	**citeseer**
GCN	0.841	0.716
SAGE	0.824	0.684
GAT	0.817	0.697

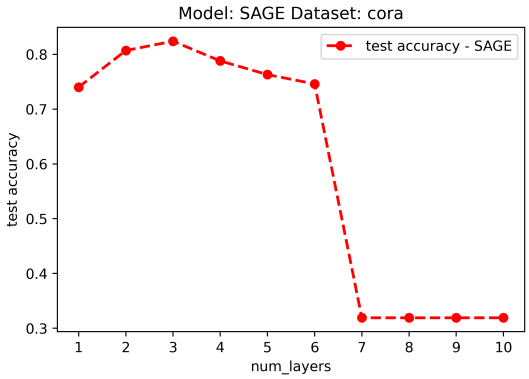
Table 2: Test accuracies with the above best hyperparameters

4 Impact on test accuracy with change in `num_layers` and `hidden_dim` hyperparameters

4.1 Change in `num_layers`



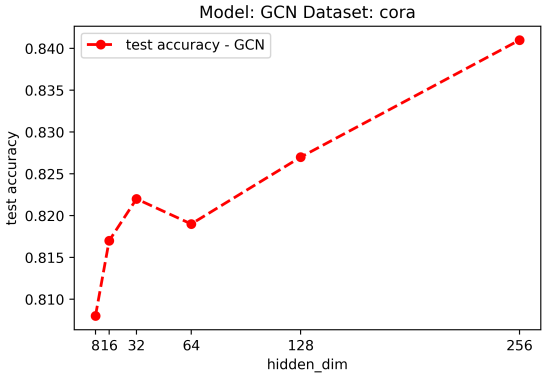
(a) GCN with increase in `num_layers`



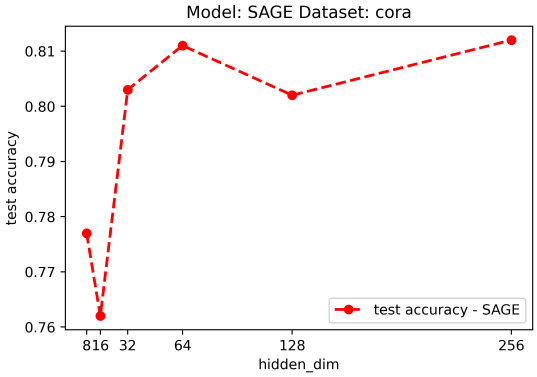
(b) SAGE with increase in `num_layers`

Figure 4: Impact on test performance of models with change in the `num_layers`.

4.2 Change in `hidden_dim`



(a) GCN with increase in `hidden_dim`



(b) SAGE with increase in `hidden_dim`

Figure 5: Impact on test performance of models with change in the `hidden_dim`.