Zakaria El Founoun
zakaria.el-founoun@student-cs.fr

Lab session # 5
ALTEGRAD 2022

12/06/22

1 Question 1

For **(1)** : For a directed graph the same approach can be applied, but this time the neighbors of the node are the ones where there exist an edge from the current node and pointed towards the neighboring node.

For **(2)** : The only modification that should be applied is to change the probability of choosing the neighbors, in fact this probability should be proportional to the edge weight.

2 Question 2

Suppose that the graph contains only nodes with no edges, then we will have:

$$\tilde{A} = I; \quad \tilde{D} = I; \quad hence, \quad \hat{A} = I$$

Then we have:

$$Z^0 = f(XW^0)$$

where f is an activation function, this is equivalent to a dense neural network for classification task.

3 Question 3

we have $Z^0 = f(\hat{A}XW^0)$, since $\tilde{A}$ is not zero only if node j is connected to node i, then Z^0 will encode information about node i and all of it's direct neighbors. Now $Z^1 = f(\hat{A}Z^0W^1)$, using the same logic and the fact that Z^0 encode information about not only the node i but also it's direct neighbors, Z^1 will encode information about the node i, the direct neighbors of i and the direct neighbors of the direct neighbors of node i (e.g distance = 2), hence the maximal distance of nodes considered in the calculation of $\hat{Y_i}$ in this GCN architecture is 2. In general case with k message passing layers, the maximal number of edges separating a given node i from the nodes the features of which are taken into account in the prediction of $\hat{Y_i}$ is k.

4 Question 4

For a complete graph:

A = $\begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}$; $\tilde{A} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$ And $\tilde{D} = 4 * I_4$ where I_4 is the identity matrix of dimension 4 * 4 hence in final we have $\hat{A} = \frac{1}{4}\tilde{A}$.

Since : $X = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$ Doing the computations we have: $Z_0 = Relu(\hat{A}.X_0.W_0) = Relu(\begin{bmatrix} -0.8 & 0.5 \\ -0.8 & 0.5 \\ -0.8 & 0.5 \\ -0.8 & 0.5 \end{bmatrix}) = \begin{bmatrix} 0 & 0.5 \\ 0 & 0.5 \\ 0 & 0.5 \\ 0 & 0.5 \end{bmatrix}$

And then

$$Z1 = Relu(\hat{A}.Z_0.W_1) = Relu(\begin{bmatrix} -0.2 & 0.3 & 0.25 \\ -0.2 & 0.3 & 0.25 \\ -0.2 & 0.3 & 0.25 \\ -0.2 & 0.3 & 0.25 \end{bmatrix}) = \begin{bmatrix} 0 & 0.3 & 0.25 \\ 0 & 0.3 & 0.25 \\ 0 & 0.3 & 0.25 \\ 0 & 0.3 & 0.25 \end{bmatrix}$$

For a star graph:

$$\hat{A} = \begin{bmatrix} 0.25 & 0.35 & 0.35 & 0.35 \\ 0.35 & 0.5 & 0 & 0 \\ 0.35 & 0 & 0.5 & 0 \\ 0.35 & 0 & 0 & 0.5 \end{bmatrix},$$

hence : $Z_0 = \begin{bmatrix} 0 & 0.65 \\ 0 & 0.42 \\ 0 & 0.42 \\ 0 & 0.42 \end{bmatrix}$

I final we have $Z_1 = \begin{bmatrix} 0 & 0.36 & 0.30 \\ 0 & 0.26 & 0.22 \\ 0 & 0.26 & 0.22 \\ 0 & 0.26 & 0.22 \end{bmatrix}$

We notice that Z_1 has always a zero entry in it's first row and that, the representation of each node depends only on it's degree.
If we randomly sampled node features X from a random uniform distribution, the structure we had will vanish and nodes with the same degree will likely have different representations.