KALLEL Mahdi
mahdi.kallel@telecom-paris.fr

1 Question 1

In the same connected component, if we look at the nodes within a same community have the same embedding. So if we were to perform spectral clustering on the community, we will find that the embeddings within each cluster are pretty close. This is because word2vec creates the embedding for each node based on it's context (which in our case is the neighbouring nodes in the generated walks).
Random walks within the same community will tend to generate approximately the same context. And thus approximately the same embeddings.
Now if we look at the embeddings of each connected component. We should find that the embeddings of component 1 form a sperate ball from those of component 2.
This is because by definition any node from C1 does not share any context with a node from C2.

2 Question 2

Although nodes v1 and v2 are structurally identical, they should not have the same embeddings.
This is because DeepWalk works by using the context of the node (which in this case in the "number" of each node in a random walk).
For example sake, if we take the context to be the 3 neighbouring nodes in a random walk of length 5. Then the context of of v1 will usually contain nodes [2,3,4] whereas that of v2 will usually contain nodes [5,6,7]. Therefore the embeddings for these nodes should be quite different.

One idea to get nodes with the same structure to have the same embedding would be to run a number "k" of **Weisfeiler-Lehman Test of Isomorphism** and instead of using the number of each node as a label, we use the number obtained by this algorithm.
This way we ensure that the context of a node is more dependant on the structural function of it's neighbours, rather than just their "number".

3 Question 3

$$\hat{A} = \begin{bmatrix} \frac{1}{\sqrt{3}} & & & \\ & \frac{1}{\sqrt{3}} & & \\ & & \frac{1}{\sqrt{3}} & \\ & & & \frac{1}{\sqrt{3}} \end{bmatrix} \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{3}} & & & \\ & \frac{1}{\sqrt{3}} & & \\ & & \frac{1}{\sqrt{3}} & \\ & & & \frac{1}{\sqrt{3}} \end{bmatrix} = (\frac{1}{3}) * \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix}$$

$$XW^1 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} 0.8 & -0.5 \end{bmatrix} = \begin{bmatrix} 0.8 & -0.5 \\ 0.8 & -0.5 \\ 0.8 & -0.5 \\ 0.8 & -0.5 \end{bmatrix}$$

$$\tilde{A}XW^1 = \begin{bmatrix} 0.8 & -0.5 \\ 0.8 & -0.5 \\ 0.8 & -0.5 \\ 0.8 & -0.5 \end{bmatrix}$$

$$Z^1 = \begin{bmatrix} 0.8 & 0 \\ 0.8 & 0 \\ 0.8 & 0 \\ 0.8 & 0 \end{bmatrix}$$

$$\tilde{A}Z^1W^2 = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0.8 & 0 \\ 0.8 & 0 \\ 0.8 & 0 \\ 0.8 & 0 \end{bmatrix} \begin{bmatrix} 0.9 & -1,2 & 0.4 \\ 1.4 & -0.3 & 0.2 \end{bmatrix}$$

$$\frac{1}{3} * \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0.72 & -0.96 & 0.36 \\ 0.72 & -0.96 & 0.36 \\ 0.72 & -0.96 & 0.36 \\ 0.72 & -0.96 & 0.36 \end{bmatrix} = \begin{bmatrix} 0.72 & -0.96 & 0.36 \\ 0.72 & -0.96 & 0.36 \\ 0.72 & -0.96 & 0.36 \\ 0.72 & -0.96 & 0.36 \end{bmatrix}$$

$$Z^2 = \begin{bmatrix} 0.72 & 0 & 0.36 \\ 0.72 & 0 & 0.36 \\ 0.72 & 0 & 0.36 \\ 0.72 & 0 & 0.36 \end{bmatrix}$$

We notice that the embeddings for each node are the same.

Since we are dealing with a cyclical graph, all the nodes have the same structural properties.
Unlike DeepWalk, graph neural networks are able to yield perfectly the same embedding for nodes that occupy the same structure.

4 Question 4

When we make the feature ”input” matrix to be the identity matrix, we reach a test accuracy of 1 on the test karate dataset.
If we change the features to be a matrix of all ones, the test accuracy is 0.47, meaning that our model is unable to learn.
We hypothesize that the initial feature matrix should have some what different initial representations for each node in order for GNN to work.
Since the models inner embeddings of a node are a function of the adjacency matrix, if different nodes have the same initial feature representation but different adjacency (ex one has two neighbours and the other three). Then we eventually get different inner representations of each node.

We believe, in the case of having the same initial representation for each node, many more message passing layers are required in order to get a good embedding and thus be able to separate the classes in the dataset.

5 Plots

:
We report our plots in the following section.

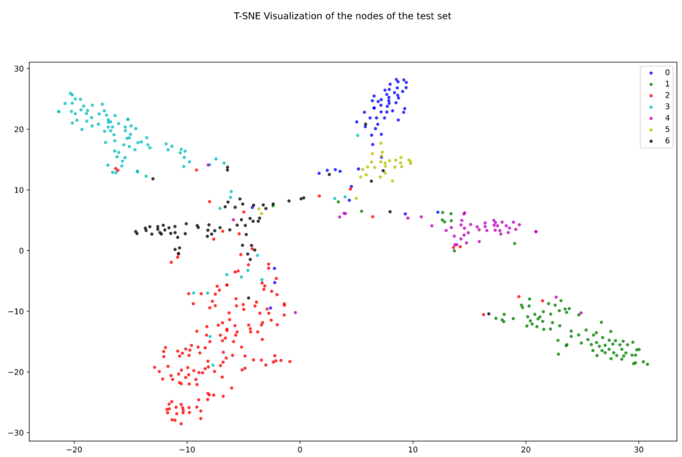

Figure 1: GNN embeddings of the classes of the cora dataset

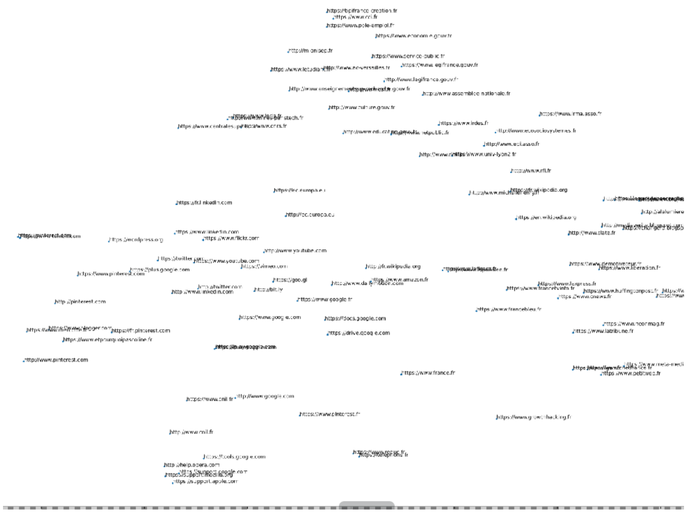

Figure 2: DeepWalk embeddings of the french WEB

References