Del Gallo Adriano
adriano.delgallo@telecom-paristech.fr

Lab session # 6
ALTEGRAD 2019

12/31/19

1 Question 1

We modify the random walks generator. If we are in v_i, to avoid $v_{i+1} = v_{i-1}$ we can select v_{i+1} uniformly among the the set $\mathcal{N}(v_i) \setminus v_{i-1}$. If this set is empty, we just stop the chain and pad with specific token.

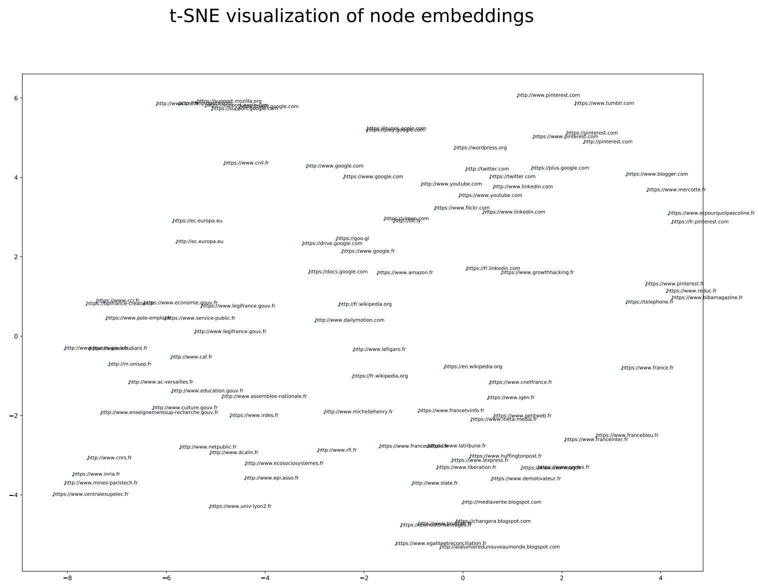

Figure 1: T-SNE visualisation of french web

2 Question 2

The 2 message passing layers give us a feature representation of each node so by adding a layer where we combine those features to obtain a representation for the entire graph before submitting it to the fully connected layer and softmax, we will be able to perform classification on graphs. This recombination can be done by summing or averaging the node vectors which would be the naïve approach.

We can also propose another approach where we can take advantage of the fact that the output of the first message passing layer will embed information about the 1-hop neighborhood, that the second one will contain properties of the 2-hop neighborhood, etc. This will enable us to build a very rich feature vector containing information on different sizes of the neighborhood for each node (as in lab 3 for text classification). We can then concatenate this representation and feed it to FC + Softmax layers. To reduce model complexity, [1] propose to use a SortingPool layer to order the most important nodes in the graph and apply other Convolution and MaxPooling layers to reduce the size of the feature vectore before giving it as input to a FC layer + Softmax.

3 Question 3

The GNN outperformed the Deepwalk + logistic regression by achieving 100% accuracy on the karate dataset compared to 85%. This is because the embeddings given by the message passing layers contains information about all the nodes and their 2-hop neighborhoods. Furthermore more, this approach can be linked to the Weisfeiler-Lehman subtree kernel, which is known to be a state of the art method to extract multi-scale subtree features. Thus the GNN aproach uncovers complex structures of the graphs to produce high quality features.

Deepwalk embeddings on the other hand give us a good idea about how information flows in the graph but fails to capture the multi scales structures within it, which is why it was outperformed. However, the Deepwalk approach did sometimes give 100% accuracy. The fact that most of time it achieved only 0.85 might be the effect of the noise mentioned in Question 1 which will cause greater problems on small graphs. Fixing this approach may help Deepwalk catch up to GNN.

Figure 2: T-SNE representation of node features in cora dataset

4 Question 4

The performance dropped significantly when we use a feature matrix with only ones to 0.28. This is due to the fact that all the nodes share a common feature vector causing the model to consider that all the nodes are the same, thus leading to obvious misclassification.

References

[1] Marion Neumann Muhan Zhang, Zhicheng Cui and Yixin Chen. An end-to-end deep learning architecture for graph classification. In *In Proceedings of the 32nd AAAI Conference on Artificial Intelligence*, page 4438–4445, 2018.