Proposal: A Feasible Solution to Embedding Large-Scale Graphs by Decomposition and its Application to Traffic Condition Prediction

Yuyang Liu
10660111

December 3, 2020

Keywords— Large-Scale Graph Embedding, Graph Decomposition, Unsupervised Clustering, Spatio-Temporal Data, Traffic Condition Prediction

1 Motivation

This project is proposed due to the problem we met while took part in a competition hold by China Computer Federation (CCF), where we were given real historical data of some roads in China and corresponding information of those roads to predict the traffic condition of a target road in a target time slice. The problem is that because of the sensibility of the data, longitude and latitude of each road were erased, merely the upstream and downstream relationships are provided. We tried to use the topological information by encoding the nodes of the graph into vectors, however, since the graph contains more than 680,000 nodes, traditional graph embedding algorithms such as DeepWalk [4] and Node2Vec [1] can't be directly applied (on our device) due to their time and space complexity. One way to solve this problem is decomposing the graph into smaller graphs and train the embeddings independently. As mentioned before, the coordinates of each link is unknown in this dataset, meaning that we can't simply split the graph based on its vertices' location in the real word. Certain novel algorithms such as PyTorch-BigGraph [2] tried to split adjacent matrix into smaller blocks, however, we doubt that their graph partition methods may cause information loss (especially in traffic prediction), and can be further improved. Hence, in this project, we seek to find a graph partition method that better reserves the structural information in a road network, and will experiment it with various existing models and algorithms.

2 Background Information

Traffic condition prediction is a composite problem, where "composite" means it requires modeling both spatial and temporal dependencies at the same time [6]. Temporal data include the historical information of roads (also called links) such as average car speed of each time slice, while spatial data contain relatively static information such as the attributes of each link and the spatio structure of a road network. One key challenge in this domain is how to use the topological information of a map or a road network. In practice, a road network can be represented in various formats, such as adjacent matrix, adjacent list, list of edges, list of links or their combinations. Although these formats can be understood by humans, they can't be directly fed into machine learning models. To use the topological information, we need to encode them first. DeepWalk [4] is a well-known graph embedding algorithm where many other methods such as Node2Vec [1] derived from. It assumes that a node in a graph is similar to a word in a corpus, and hence can be represented by co-occurrence among nodes. This algorithm uses a deep-first-search based random-walk algorithm to traverse a graph, and treats the result as a document in a corpus. Then it can apply skip-gram model [3] on the corpus to get the embedding of each node. LINE [5] and Node2Vec [1] follow the idea but use different searching algorithm

instead. The key step of those algorithms is how to building a corpus from a graph. For each node, it needs to be chosen as the start point of a traverse at least once to ensure all nodes are converted into embeddings. However, when the graph is too large, for example, in the dataset mentioned above, this method could bring problems to storage and computation. To apply these algorithms on such a graph, we need to partition the graph first, and this brings new questions:

1. follow which rule to split the graph?
2. should the graph be partitioned into smaller graphs with equal size or different size?
3. will the difference of size impact the embedding results if the graph is not partitioned equally?
4. how large each small graph should be if the graph is partitioned equally?
5. can graph embeddings correctly reflect the relationships between nodes sharing similar spatio structure in a road network when they are trained independently?
6. will this method can bring improvement to real word application?

3 Research Scope

Graphs can be grouped into different categories, such as digraph and undigraph, connected and unconnected graph, weighted and unweighted graph, and they are widely used in artificial intelligence researches such as knowledge graph. In this project, we won't investigate them all, instead, we will mainly focus on the one mentioned in section 1, which is an undirected, weighted and connected graph (provided that each road is real existing, and connected to other roads). In addition, we will merely experiment on traffic prediction tasks and corresponding datasets.

4 Aims and Objectives

The general aims of this projects are finding a feasible large-scale graph embedding method and applying it to traffic prediction tasks. They can be detailed as follow:

1. each node should have an corresponding vector (either unique or not) that can reflects the structural relationship with other nodes.
2. the vectors should be integrated with existing models or algorithms in certain methods.

The key objectives of this project are:

1. re-format a graph represented by upstream and downstream list.
2. build a relatively smaller graph for experiment based on the re-formated graph if necessary.
3. experiment different unsupervised clustering algorithms to split a large graph into small graphs, and apply graph embedding algorithms to them.
4. set up criteria for the evaluation of the quality of generated embeddings
5. revise the clustering algorithms or design a new one based on experimental results and the intrinsic properties of road network.
6. build and train several deep learning models integrated with and without the embeddings produced by our algorithm respectively, and compare their performance.

References

[1] Grover, A., and Leskovec, J. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining* (2016), pp. 855–864.

[2] Lerer, A., Wu, L., Shen, J., Lacroix, T., Wehrstedt, L., Bose, A., and Peysakhovich, A. Pytorch-biggraph: A large-scale graph embedding system. *arXiv preprint arXiv:1903.12287* (2019).

[3] Mikolov, T., Chen, K., Corrado, G., and Dean, J. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[4] Perozzi, B., Al-Rfou, R., and Skiena, S. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining* (2014), pp. 701–710.

[5] Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., and Mei, Q. Line: Large-scale information network embedding. In *Proceedings of the 24th international conference on world wide web* (2015), pp. 1067–1077.

[6] Yao, H., Tang, X., Wei, H., Zheng, G., and Li, Z. Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence* (2019), vol. 33, pp. 5668–5675.