Papers' Review

Stefanos Antaris

KTH Royal Institute of Technology

Over the last few years, graph neural networks have been proven beneficial in several real-world applications that could be modeled as graphs. The main goal of graph neural networks is to exploit deep neural network architectures to capture the structural and temporal properties of each node of the graph in a low-dimensional node representation, namely node embeddings. Most of the existing approaches focus on static graphs, ignoring completely the dynamic nature of the real-world applications. Most recent approaches adopt neural network architectures to capture the evolution of the graph in the computed node embeddings. During this course, I have reviewed three recently published papers that provide novel solutions to address the problem of graph neural networks in a streaming model [1–3].

1 Paper 1: Streaming Graph Neural Networks

1.1 Summary

In this paper, the authors propose a graph neural network model, namely DyGNN [2]. The proposed model captures the evolution of the graph in a streaming fashion, based on the sequential information of edges. To capture the evolution over time, the authors modified the Long-Short Term Memory (LSTM) model, by introducing the time aspect in the update component of the LSTM architecture. Each node is represented by two embedding vectors, that is the source and the target vector. Therefore, the model considers directed interactions and the embeddings of each node are updated based on the direction of the interaction. Moreover, the authors consider the fact that each each interaction affects the nodes that participate in the interaction, but also their immediate neighbors. Therefore, the proposed model consists of a propagation component to update the neighbors' embeddings.

1.2 Strong Points

The strong points of this paper are the following:

- The authors modified the LSTM network in a novel way to incorporate the time interval between the interactions.
- The proposed model considers that the nodes' immediate neighbors are affected by the interaction. Therefore, the propagate component manages this update.

1.3 Weak Points

The weak points of this paper are the following:

- The authors are compared mostly with static graph neural network approaches, such as GCN, GraphSAGE. The relative improvement of the proposed approach against the second best baseline, that is GCN, is marginal. Therefore, it is not clear the benefit of applying the proposed model in real-wold applications
- The authors decided to exploit Recall20 and Recall50 which is a significantly high number of ranking. However, they do not provide results on top-5 or top-10 ranking.
- The proposed model considers only positive interactions. Therefore, the node embeddings are always affected positively of each interaction. However, in real-world applications, interactions can have negative semantic.

2 Paper 2: Learning temporal interaction graph embedding via coupled memory network

2.1 Summary

In this paper, the authors propose a dynamic graph neural network approach with memory networks, namely TigeCMN [3]. The authors exploit two memory networks to capture the temporal interaction patterns of the dynamic network. The first memory network augments memory matrices to contain the information of the interactions; the second network manages the memories stored in the memory matrices to capture the most recent node preferences. To accomplish the memory management, the proposed model exploits attention mechanism to identify the significance of each memory in the learned embedding.

2.2 Strong Points

The strong points of this paper are the following:

- The authors propose a novel graph neural network architecture which exploits two types of memory networks. This allows the learned node embeddings to contain information about interactions that were significant for the evolution of the node's preferences, while ignoring the rest of the interactions.
- The authors analyze the impact of each memory network in the performance of the computed embeddings. Based on the evaluation in both link prediction and node classification, the proposed model constantly outperforms both static and dynamic graph neural network baselines.

2.3 Weak Points

The weak points of this paper are the following:

- The authors have no results about the memory resources required to evaluate the model. Given that we have two coupled memory networks, the proposed approach should require significant resources in terms of memory and cpu to train and serve this model. Therefore, it is questionable the benefit of having such a big model in production, as it will have significant online inference latency.
- Similar to the DyGNN baseline, the proposed model considers only positive interactions.

3 Paper 3: End-to-end deep reinforcement learning based recommendation with supervised embedding

3.1 Summary

The authors present a deep reinforcement learning framework, namely EDRR, that exploits node embeddings to provide user-item recommendations [1]. First, the authors analyze the impact of different node embedding strategies in the deep reinforcement learning model. Based on their findings, they propose three architectures that train the node embeddings by combining the reinforcement learning loss signal with the recommendation system accuracy signal. Given the proposed architectures, the node embeddings are able to capture both the evolution of the user-item interactions, while achieving high recommendation ranking accuracy.

3.2 Strong Points

The strong points of this paper are the following:

- The authors present a simple but efficient model to train the node node embeddings in the user-item temporal interaction graph. Given the additional loss signal added in the reinforcement learning scheme, the authors convert the training of the model to a semi-supervised learning approach. This allows the model to capture both the short-term and long-term user preferences in a single embedding.
- The proposed approach provides stability on the learned embeddings, which is beneficial to achieve high recommendation accuracy.

3.3 Weak Points

The weak points of this paper are the following:

- Similar to the DyGNN, the authors should provide results with lower ranking, such as top-5 or top-10.
- The authors do not evaluate their proposed model against other recommender systems that exploit reinforcement learning.

References

1. Liu, F., Guo, H., Li, X., Tang, R., Ye, Y., He, X.: End-to-end deep reinforcement learning based recommendation with supervised embedding. In: WSDM. p. 384–392 (2020)
2. Ma, Y., Guo, Z., Ren, Z., Tang, J., Yin, D.: Streaming graph neural networks. In: SIGIR. p. 719–728 (2020)
3. Zhang, Z., Bu, J., Ester, M., Zhang, J., Yao, C., Li, Z., Wang, C.: Learning temporal interaction graph embedding via coupled memory networks. In: WWW. p. 3049–3055 (2020)