ST5188 Project Proposal Consultation: Applications of Graph Convolution Networks in Social Recommendation Systems

Yeo Ming Jie, Jonathan   Shi Qin   Chen Pu

27 January 2023

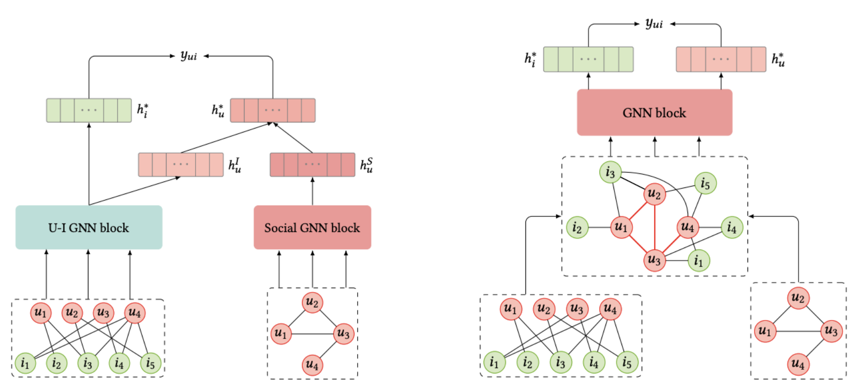
Project Proposal Overview

Recommender systems leverage on user-item interactions for the task of generating recommendations to users. Social recommendation systems are extensions to the recommender system problem, whereby user-to-user social relationships are further utilized to enhance user representations for better predictions. In this project, we propose to explore the use of Graph Convolution Networks (GCNs) as applied in the field of social recommendation systems for predicting user-item interactions. Specifically, we focus on the LightGCN model and an extension of the LightGCN model to social recommendation, SocialLGN. LightGCN is a simplified version of Neural Graph Collaborative Filtering (NGCF), which adapts GCNs in recommendation systems.

Proposed Research Directions

1. Applying LightGCN to Social Recommendation via a Unified Graph Approach

SocialLGN learns user representations from user-item interactions and social graphs separately before integrating them into a final preference vector. As inspired by the models Diffnet++ and SEFrame which adopt a hierarchical aggregation schema to integrate the two graphs into a unified network, LightGCN could be adapted to solve the social recommendation problem via a unified graph approach instead. As users play a central role in social network and interest network, jointly modeling the higher-order structure of these two networks would mutually enhance each other, thus motivating the integration of both graphs into a unified network.



(a) The framework of GNN on the bipartite graph and social network graph separately.

(b) The framework of GNN on the unified graph of user-item interactions and social network.

Fig. 7. Two strategies for social enhanced general recommendation.

2. VQ Mechanism

Vector Quantization(VQ) is widely used in the fields of signal processing and data compression. The basic principle of VQ is to use the index of the code word that best matches the input vector, so as to replace the input vector for transmission and storage. The compilation of a codebook is done with the help of clustering. In VQ-GNN, each mini-batch message passing is approximated by the process of VQ codebook update and approximated message passing. In the learning task of recommender system based on GNN, in each mini-batch and for a certain user/item, we can obtain their neighbour's code word according to the codebook before aggregating the information. This will have an effect of dimensionality reduction, thereby improving the efficiency and shortening the running time.

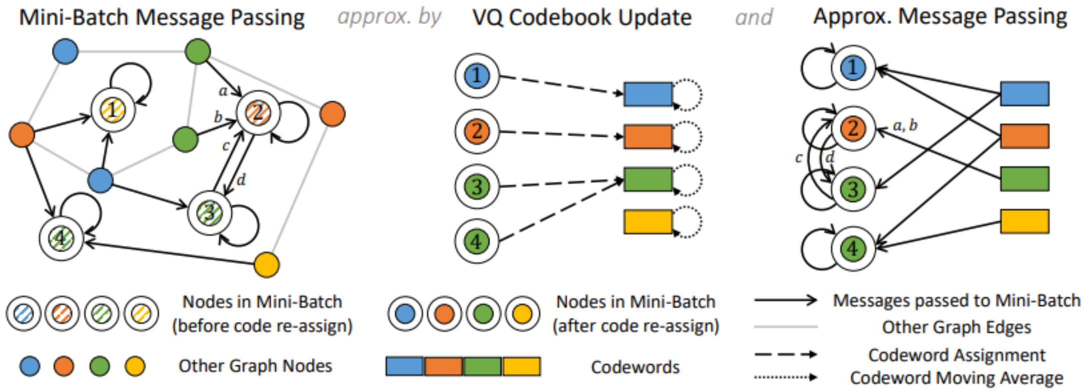


Figure 1: In our framework, VQ-GNN, each *mini-batch message passing* (left) is approximated by a *VQ codebook update* (middle) and an *approximated message passing* (right). All the messages passed to the nodes in the current mini-batch are effectively preserved. Circles are nodes, and rectangles are VQ codewords. A double circle indicates nodes in the current mini-batch. Color represents codeword assignment. During *VQ codebook update*, codeword assignment of nodes in the mini-batch is refreshed (node 1), and codewords are updated using the assigned nodes. During *approximated message passing*, messages from out-of-mini-batch nodes are approximated by messages from the corresponding codewords, messages from nodes assigned to the same codeword are merged (a and b), and intra-mini-batch messages are not changed (c and d).

3. Second-order user-item graph

In the SocialLGN model, the item representations only depends on their neighbor node of user in the user-item graph. However, a large number information about the items' connection is ignored. In the real world, the items which appears together are always positively relative. There are many methods inspired by the phenomenon and applied in the different field, such as market basket analysis and item2vec. This idea is still practical in the Social Recommendation System. The connection of items is able to be represented by the second-order user-item graph (Fig 1) and the embedding in the model, denoted as

$$q_i^{(k)} = \sum_{i \in N_i^{I_2}} \frac{1}{\sqrt{N_i^{I_2}\sqrt{N_v^S}}} q_i^{(k-1)}$$