Research Plan

Target: submit the paper in July 2025

Topic: Temporal GNN link prediction

* Continuous TLP Problem setup: (before Jan. 20)
* Problem Formulation:

A temporal network can be represented as a sequence of links that come in over time, i.e. where is a link and is the timestamp showing when arrives. Each link corresponds to a dyadic event between two nodes . For simplicity, we first assume those links to be undirected and without attributes while later we discuss how to generalized our method to directed attributed networks. The sequence of links encodes network dynamics. Therefore, the capability of a model for representation learning of temporal networks is typically evaluated by how accurately it may predict future links based on the historical links. In this work, we also use link prediction as the metric. Note that we care not only the link prediction between the nodes that have been seen during the training. We also expect the models to predict links between the nodes that has never been seen as the inductive evaluation.

* *Transductive link prediction task* allows temporal links between all nodes to be observed up to a time point during the training phase, and uses all the remaining links after that time point for testing. In our implementation, we split the total time range [0, T] into three intervals: , .
* *Inductive link prediction task* predicts links associated with nodes that are not observed in the training set. There are two types of such links: 1) "old vs. new" links, which are links between an observed node and an unobserved node; 2) "new vs. new" links, which are links between two unobserved nodes. Since these two types of links suggest different types of inductiveness, we distinguish them by reporting their performance metrics separately. In practice, we follow two steps to split the data: 1) we use the same setting of the transductive task to first split the links chronologically into training / validation / testing sets; 2) we randomly select 10% nodes, remove any links associated with them from the training set, and remove any links not associated with them in the validation and testing sets.
* randomly sample an equal amount of negative links and consider link prediction as a binary classification problem.
* Baselines:
* (1) Snapshot-based methods, including DynAERNN (Goyal et al., 2020), VGRNN (Hajiramezanali et al., 2019) and EvolveGCN (Pareja et al., 2020);
* (2) Stream-based methods, including TGAT (Xu et al., 2020), JODIE (Kumar et al., 2019) and DyRep (Trivedi et al., 2019), TGN, CAWN, TCL, GraphMixer, DyGFormer
* Datasets:
* Wikipedia, Reddit, MOOC, LastFM, Enron, Social Evo., UCI, Flights, Can. Parl., US Legis., UN Trade, UN Vote, and Contact

Discrete TLP Problem setup:

High-Quality Temporal Link Prediction for Weighted Dynamic Graphs via Inductive Embedding Aggregation

A dynamic graph can be represented as a sequence of snapshots over time steps . Each snapshot can be described as , where is the node set; is the weighted edge set. We assume that attributes of each node are fixed for all the snapshots, but the topology structure (including the node and edges sets) can change over time. In particular, the fixed node attributes can be information denoting the unique identity of each node in a system (e.g., IP address).

We use an adjacency matrix to describe the topology of each snapshot , where and otherwise. Moreover, we use an attribute matrix to describe node attributes of , where the i-th row is the attribute vector of node . For each time step t, we also use an aligning matrix to encode the node index correspondence between two successive snapshots In each snapshot , node indices are renumbered from 1. if node corresponds to and otherwise (if we allow nodes or edges change with time). In particular, is an identity matrix when all the snapshots share a common node set.

Question 1: we assume that all the snapshots share a common node set V, i.e., . Namely, there is no addition and deletion of nodes. We also have , with X as a common attribute matrix for all the snapshots. Given the topology of previous l snapshots described by and attributes described by X, TLP aims to predict the topology of next time step (τ + 1) described by .

Question 2: we assume that different snapshots can have different node sets. In particular, we only focus on the prediction of weighted edges between nodes observed in previous snapshots. For simplicity, we define be the attribute matrix w.r.t. nodes in Given the topology and attributes of previous snapshots (described by , , and ) as well as the node set and attributes described by and , TLP aims to predict the topology of next time step w.r.t. .

Question 3: not only predict edges between nodes in , but also predict edges (i) between a previously observed node and an unobserved node or (ii) between two unobserved nodes using the prior knowledge of Given the topology and attributes of previous snapshots (described by , and ) as well as the node set and attributes of next time step (described by and ).

Datasets:

Network dataset: Mesh, Hmob,

Traffic dateset: DC, T-Drive,

Social dataset: SEvo,

Internet: IoT, WIDE,

Baseline:

CRJMF, DeepEye, TMF, LIST, D2V, DDNE, E-LSTM-D, EGCN, DySAT, STGSN, GCN-GAN, NetGAN

Existing work (Jan. 20-Mar. 7)

* Familiar with existing work related to GNN link prediction (MC)
* Familiar with existing work related to temporal GNN link prediction (MC)
* Run baselines on datasets (MC)
* discuss the method (all)
* Applying HL-GNN into temporal GNN (Mar. 7-Apr 30)
* Summarize the contributions of applying HL-GNN into temporal GNN (all)
* Design model that applies HL-GNN into temporal GNN for link prediction and exp (MC)
* Think about how to apply the theoretical of HL-GNN to temporal GNN (YN)
* Theoretical analysis (May 1-June 1)
* Paper writing (June 1-July 1)

Risk: (1) Extending static GNN to dynamic may be difficult in theoretical analysis. (2) The coding task is heavy and may exceed the expected time. Consider thinking about the theoretical analysis part in the third stage and compressing the time for theoretical analysis and writing.