Graph Neural Networks for temporal graphs: State of the art, open challenges, and opportunities

https://doi.org/10.48550/arXiv.2302.01018

2024-09-27

Static vs. Temporal

- Static graph: V, E, Xv, Xe
- Temporal graph: time-dependent node and edge feature

$$V_T := \{(v, x^v, t_s, t_e) : v \in V, x^v \in \mathbb{R}^{d_V}, t_s \leq t_e\},$$
$$E_T := \{(e, x^e, t_s, t_e) : e \in E, x^e \in \mathbb{R}^{d_E}, t_s \leq t_e\},$$

Temporal representation

- Snapshot-based:
  - focuses on the temporal evolution of the whole graph.

$$G_T^S := \{(G_i, t_i) : i = 1, \ldots, n\}$$

- Event-based:
  - Event of node/edge insert/delete

Snapshot-based Graph:Model Evolution methods

- EvolveGCN
  - A RNN to update network parameters
  - Variable node size

Snapshot-based Graph: Embedding Evolution methods

- DySAT
  - Two layer of self-attention, one for get embedding from graph at each time step, one for get new embedding from historical embedding
  - Flexibility
- VGRNN
  - Variational Graph auto-encoder to get embedding
  - Filtered
  - LSTM to get next embedding

Snapshot-based Graph: Embedding Evolution methods cont.

- ROLAND
  - Idea: embedding at each GNN layer is hierarchical node state
  - hierarchical node state updated by gated recurrent unit (GRU)
- DynGESN
  - Node embedding by previous node embedding and temporal neighboorhood
- SSGNN
  - trainable parameters in the decoder
  - representations of the time series data of each node