Heterogeneous Graph Transformer

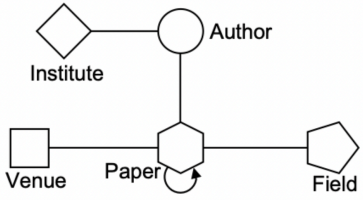
Ziniu Hu, Yuxiao Dong, Kuansan Wang, Yizhou Sun

Jun 03, 2022

Outline

Introduction

Heterogeneous Graph (HG) also known as heterogeneous information networks (HIN). A heterogeneous graph can represent as $\mathcal{G} = (\mathcal{V}, \xi)$, where each node $\mathcal{V}$, and each edge ξ has its own type Γ_v and Γ_e. A heterogeneous graph have two mapping function: $\phi_v : V \rightarrow \Gamma_v$ for node to node types, and $\phi_e : \xi \rightarrow \Gamma_e$ for edge types.



(a) The schema of heterogeneous academic networks

Author	*is_(first/last/other)_author_of*	Paper
Author	*is_affiliated_with*	Institute
Paper	*is_published_(conf/journal)_at*	Venue
Paper	*has_(L_1−L_5)_field_of*	Field
Paper	*has_citation_to*	Paper

(b) The meta relations of heterogeneous academic networks

Problem

- Meta-path need domain knowledge.
- Different types of nodes/edges share features.
- Different types of nodes/edges keep different non-shared weights
- Ignore the dynamic of heterogeneous graph
- Incapable of modeling Web-scale (large) heterogeneous graph

Heterogeneous graph transformer (HGT)

- Node and edge type dependent attention mechanism.
  - Not parameterizing each type of edges
  - use meta relation triplet $e = (s, t)$, where s is source node, t is target node
- Relative temporal encoding (RTE) strategy for dynamic graph
- HGSampling for Web-scale graph data.

Symbols

Graph	$G = (\mathcal{V}, \mathcal{E}, \mathcal{A}, \mathcal{R})$
Node	$v \in \mathcal{V}$, also s, t
Edge	$e \in \mathcal{E}$
Node Type	$\tau(v) : \mathcal{V} \to \mathcal{A}$
Edge Type	$\phi(e) : \mathcal{E} \to \mathcal{R}$
edge, source node, target node	$e = (s, t)$
meta relation triplet	$< \tau(s), \phi(e), \tau(t) >$

Method

Use the meta-relations fo heterogeneous graph to parameterize weight matrices for heterogeneous mutual attention, message passing, and propagation steps.
Three steps:

- Heterogeneous Mutual Attention
  - input embedding of s_1, s_2, t
  - output attention matrix of $\phi(e)$.
- Heterogeneous Message Passing
  - output message of $\phi(e)$
- Target-Specific Aggregation

Method

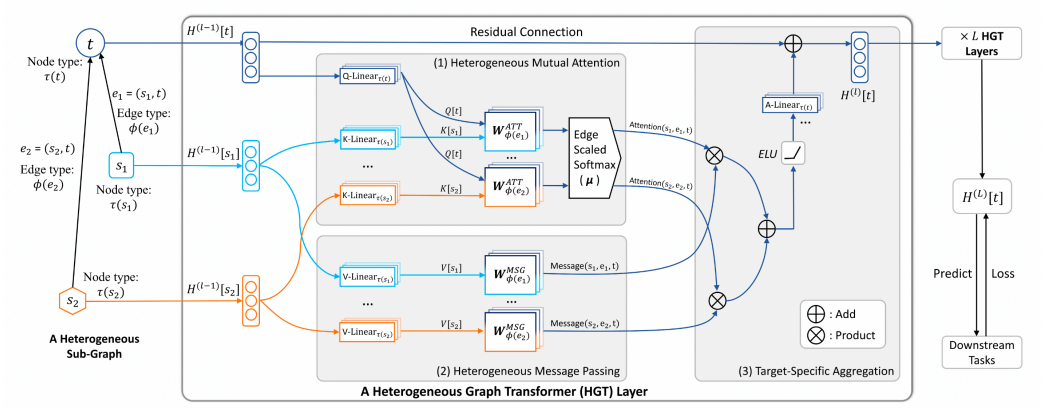


Figure 2: The Overall Architecture of Heterogeneous Graph Transformer. Given a sampled heterogeneous sub-graph with t as the target node, s_1 & s_2 as source nodes, the HGT model takes its edges $e_1 = (s_1, t)$ & $e_2 = (s_2, t)$ and their corresponding meta relations $< \tau(s_1), \phi(e_1), \tau(t) >$ & $< \tau(s_2), \phi(e_2), \tau(t) >$ as input to learn a contextualized representation $H^{(L)}$ for each node, which can be used for downstream tasks. Color decodes the node type. HGT includes three components: (1) meta relation-aware heterogeneous mutual attention, (2) heterogeneous message passing from source nodes, and (3) target-specific heterogeneous message aggregation.

Heterogeneous Mutual Attention

GAT:

$$H^l[t] \leftarrow \underset{\forall s \in N(t), \forall e \in E(s,t)}{\textbf{Aggregate}} \Big(\textbf{Attention}(s,t) \cdot \textbf{Message}(s) \Big) \quad (2)$$

$$\begin{aligned}
\textbf{Attention}_{GAT}(s,t) &= \underset{\forall s \in N(t)}{\text{Softmax}} \Big(\vec{a} \Big(W H^{l-1}[t] \parallel W H^{l-1}[s] \Big) \Big) \\
\textbf{Message}_{GAT}(s) &= W H^{l-1}[s] \\
\textbf{Aggregate}_{GAT}(\cdot) &= \sigma \Big(\text{Mean}(\cdot) \Big)
\end{aligned}$$

Attention Importance of each source node.

Message Extracts the message by using only the source node.

Aggregate Aggregate the neighborhood message by the attention weight.

Heterogeneous Mutual Attention

Transformer: W_q, W_k, W_v
HGT:

$$\textbf{Attention}_{HGT}(s,e,t) = \underset{\forall s \in N(t)}{\text{Softmax}}\Big(\Big\|_{i\in[1,h]} ATT\text{-}head^i(s,e,t)\Big) \quad (3)$$

$$ATT\text{-}head^i(s,e,t) = \Big(K^i(s)\, W^{ATT}_{\phi(e)}\, Q^i(t)^T\Big) \cdot \frac{\mu_{\langle \tau(s),\phi(e),\tau(t)\rangle}}{\sqrt{d}}$$

$$K^i(s) = \text{K-Linear}^i_{\tau(s)}\Big(H^{(l-1)}[s]\Big)$$

$$Q^i(t) = \text{Q-Linear}^i_{\tau(t)}\Big(H^{(l-1)}[t]\Big)$$

- $W^{ATT}_{\phi(e)}$
- $\mu_{<\tau(s),\phi(e),\tau(t)>}$

Message passing

$$\textbf{Message}_{HGT}(s,e,t) = \Big\Vert_{i\in[1,h]} MSG\text{-}head^i(s,e,t)$$

$$MSG\text{-}head^i(s,e,t) = \text{M-Linear}^i_{\tau(s)}\Big(H^{(l-1)}[s]\Big)\, W^{MSG}_{\phi(e)}$$

- Edge dependent: $W^{(MSG)}_{\tau(e)}$
- Incorporate the meta relations of edges into the message passing process to alleviate the distribution differences of nodes and edges of different types.

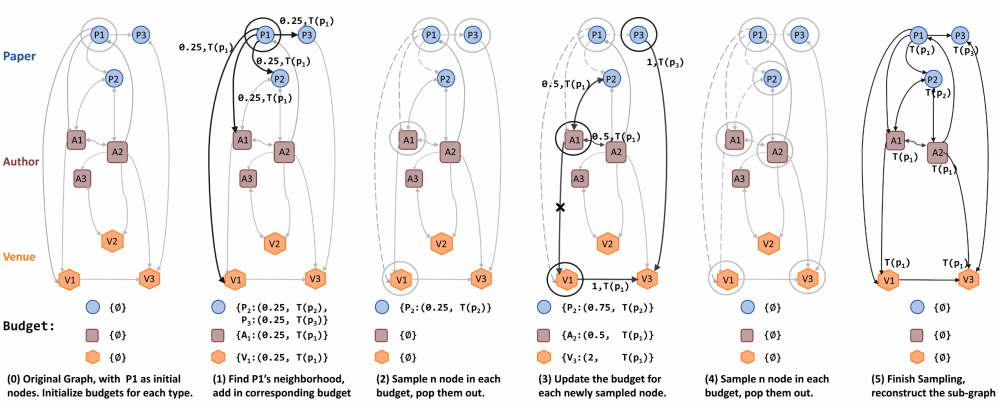
Target-Specific Aggregation

$$\widetilde{H}^{(l)}[t] = \bigoplus_{\forall s \in N(t)} \Big(\textbf{Attention}_{HGT}(s, e, t) \cdot \textbf{Message}_{HGT}(s, e, t) \Big).$$

$$H^{(l)}[t] = \text{A-Linear}_{\tau(t)} \Big(\sigma \big(\widetilde{H}^{(l)}[t] \big) \Big) + H^{(l-1)}[t]. \tag{5}$$

- A-Linear$_{\tau(t)}$ to map target node t to type specific distribution and update the l-th HGT layers embedding.

HGSampling



- keep a similar number of nodes and edges for each type, and keep the sampled sub-graph dense to minimize the information loss and reduce the sample variance.

Relative Temporal Encoding (RTE)

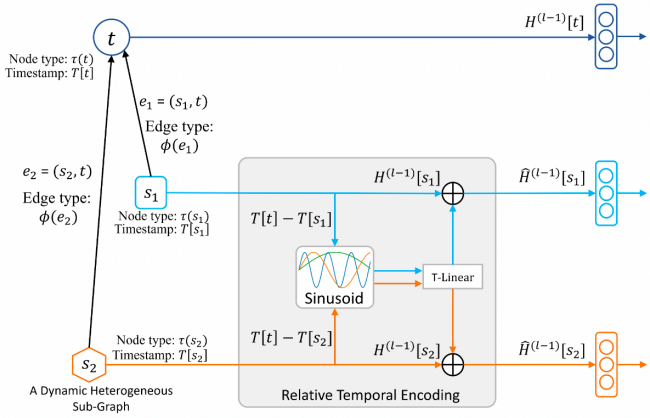


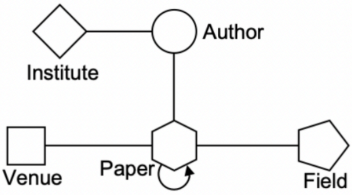
Figure 3: Relative Temporal Encoding (RTE) to model graph dynamic. Nodes are associated with timestamps $T(\cdot)$. After the RTE process, the temporal augmented representations are fed to the HGT model.

Relative Temporal Encoding (RTE)

$$\widehat{H}^{(l-1)}[s] = H^{(l-1)}[s] + RTE\big(\Delta T(t,s)\big) \tag{9}$$

- $\Delta T(s,t) = T(s) - T(t)$

OAG Data



(a) The schema of heterogeneous academic networks

Author	*is_(first/last/other)_author_of*	Paper
Author	*is_affiliated_with*	Institute
Paper	*is_published_(conf/journal)_at*	Venue
Paper	*has_(L_1-L_5)_field_of*	Field
Paper	*has_citation_to*	Paper

(b) The meta relations of heterogeneous academic networks

Figure 1: The schema and meta relations of Open Academic Graph (OAG). Given a Web-scale heterogeneous graph, e.g., an academic network, HGT takes only its one-hop edges as input without manually designing meta paths.

OAG Data

- OAG
  - All
  - Computer Science (CS)
  - Medicine (Med)

Baseline models

- Graph Convolutional Networks (GCN)
- Graph Attention Networks (GAT)
- Relational Graph Convolutional Networks
  - Keep a different weight for each relationship (edge).
  - $h_i^{(l+1)} = \sigma\left(\sum_{r\in\mathcal{R}}\sum_{j\in\mathcal{N}_i^r}\frac{1}{c_{i,r}}W_r^{(l)}h_j^{(l)} + W_0^{(l)}h_i^{(l)}\right)$
- Heterogeneous Graph Neural Networks
  - Adopt different BiLSTM for node type and neighbor information
- Heterogeneous Graph Attention Networks (HAN)
  - Hierarchical attentions to aggregate neighbor via meta-paths

Results

GNN Models			GCN [9]	RGCN [14]	GAT [22]	HetGNN [27]	HAN [23]	HGT$^{-RTE}_{-Heter}$	HGT$^{+RTE}_{-Heter}$	HGT$^{-RTE}_{+Heter}$	HGT$^{+RTE}_{+Heter}$
# of Parameters			1.69M	8.80M	1.69M	8.41M	9.45M	3.12M	3.88M	7.44M	8.20M
Batch Time			0.46s	1.24s	0.97s	1.35s	2.27s	1.11s	1.14s	1.48s	1.50s
CS	Paper–Field (L_1)	NDCG	.608±.062	.603±.065	.622±.071	.612±.063	.618±.058	.662±.051	.689±.042	.705±.036	**.718±.014**
		MRR	.679±.069	.683±.056	.694±.065	.689±.060	.691±.051	.751±.036	.779±.027	.799±.023	**.823±.019**
	Paper–Field (L_2)	NDCG	.344±.021	.322±.053	.357±.058	.346±.071	.352±.051	.362±.048	.371±.043	.379±.047	**.403±.041**
		MRR	.353±.053	.340±.061	.382±.057	.373±.051	.388±.065	.394±.072	.397±.064	.414±.076	**.439±.078**
	Paper–Venue	NDCG	.406±.081	.412±.076	.437±.082	.431±.074	.449±.072	.456±.069	.461±.066	.468±.074	**.473±.054**
		MRR	.215±.066	.216±.105	.239±.089	.245±.069	.254±.074	.258±.085	.265±.090	.275±.089	**.288±.088**
	Author Disambiguation	NDCG	.826±.039	.835±.042	.864±.051	.850±.056	.859±.053	.867±.048	.875±.046	.886±.048	**.894±.034**
		MRR	.661±.045	.665±.054	.694±.052	.668±.061	.688±.049	.703±.036	.712±.032	.727±.038	**.732±.038**
Med	Paper–Field (L_1)	NDCG	.560±.056	.571±.061	.584±.076	.598±.068	.607±.054	.654±.048	.667±.045	.683±.037	**.709±.029**
		MRR	.465±.055	.470±.082	.493±.069	.509±.054	.575±.057	.620±.066	.642±.062	.659±.055	**.688±.048**
	Paper–Field (L_2)	NDCG	.334±.035	.337±.051	.344±.063	.342±.048	.350±.059	.359±.053	.365±.047	.374±.050	**.384±.046**
		MRR	.337±.061	.343±.063	.370±.058	.373±.061	.379±.052	.385±.071	.397±.069	.408±.071	**.417±.074**
	Paper–Venue	NDCG	.377±.059	.383±.062	.388±.065	.412±.057	.416±.068	.421±.083	.432±.078	**.446±.083**	.445±.085
		MRR	.211±.045	.217±.058	.244±.091	.259±.072	.271±.056	.277±.081	.282±.085	.288±.074	**.291±.062**
	Author Disambiguation	MRR	.776±.042	.779±.048	.828±.044	.824±.058	.834±.056	.838±.047	.844±.041	.864±.043	**.871±.040**
		NDCG	.614±.051	.625±.049	.663±.046	.659±.061	.667±.053	.683±.055	.691±.046	.708±.041	**.718±.043**
OAG	Paper–Field (L_1)	NDCG	.508±.141	.511±.128	.534±.103	.543±.084	.544±.096	.571±.089	.578±.086	.595±.089	**.615±.084**
		MRR	.556±.136	.565±.105	.610±.096	.616±.076	.622±.092	.649±.081	.657±.078	.675±.082	**.702±.081**
	Paper–Field (L_2)	NDCG	.318±.074	.328±.046	.339±.049	.336±.062	.342±.051	.350±.045	.354±.046	.358±.052	**.367±.048**
		MRR	.322±.067	.332±.052	.348±.045	.350±.053	.358±.049	.362±.057	.369±.058	.371±.064	**.378±.071**
	Paper–Venue	NDCG	.302±.066	.313±.051	.317±.057	.309±.071	.327±.062	.334±.058	.341±.059	.353±.064	**.355±.062**
		MRR	.194±.070	.193±.047	.196±.052	.192±.059	.214±.067	.229±.061	.233±.060	.243±.048	**.247±.061**
	Author Disambiguation	NDCG	.738±.042	.755±.048	.797±.044	.803±.058	.821±.056	.835±.043	.841±.041	.847±.043	**.852±.048**
		MRR	.612±.064	.619±.057	.645±.063	.649±.052	.660±.049	.668±.059	.674±.058	.683±.066	**.688±.054**

Table 2: Experimental results of different methods over the three datasets.

Futures

- Generate heterogeneous graphs
  - predict new papers and title
- Pre-train HGT to benefit tasks with scarce labels

—

- Downstream Tasks