Heterogeneous Graph Neural Network on Semantic Tree

Mingyu Guan[1], Jack W. Stokes[2], Qinlong Luo[2], Fuchen Liu[2], Purvanshi Mehta[3], Elnaz Nouri[2], Taesoo Kim[1]

[1]Georgia Institute of Technology, [2]Microsoft Corporation, [3]Lica World Inc

The 39th Annual AAAI Conference on Artificial Intelligence
FEBRUARY 25 – MARCH 4, 2025 | PHILADELPHIA, PENNSYLVANIA, USA

Introduction

- The recent past has seen an increasing interest in Heterogeneous Graph Neural Networks (HGNNs), since many **real-world graphs are heterogeneous in nature**, from citation graphs to email graphs.
- However, existing methods ignore a **tree hierarchy among metapaths**, naturally constituted by different node types and relation types.
- We present **HetTree**, a novel HGNN that builds a **semantic tree** data structure to capture the hierarchy among metapaths.
- Our evaluation demonstrates **superior performance and scalability** of HetTree on a variety of real-world graphs.

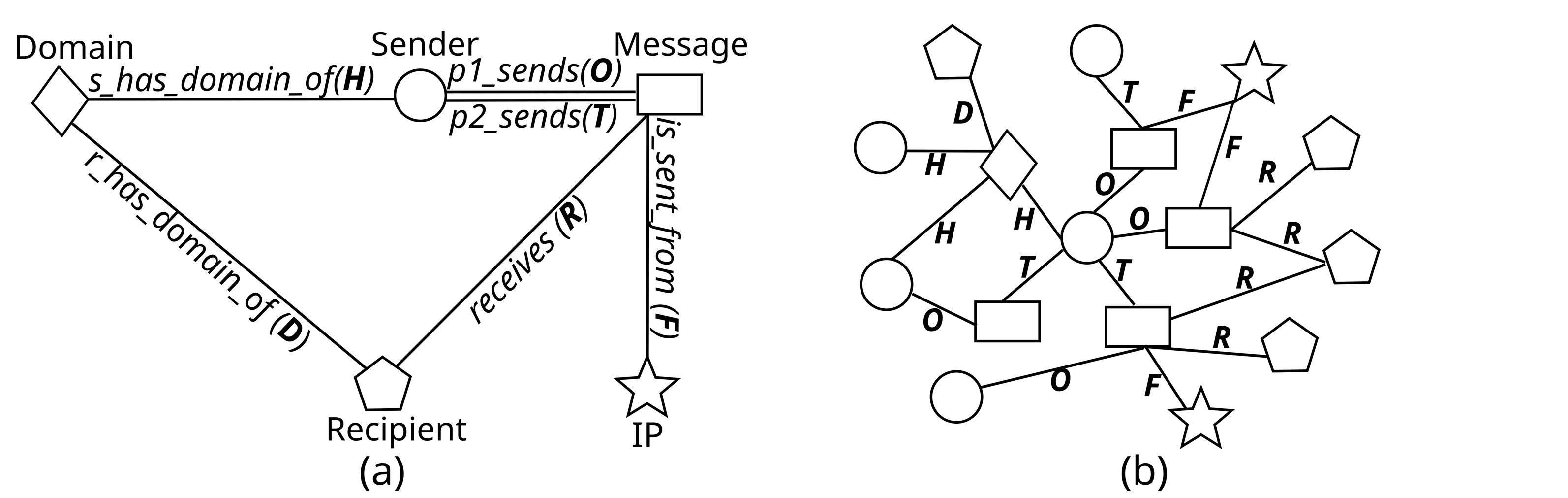

Figure 1. (a) Relational scheme of a heterogeneous email graph (b) An example of the email graph.

Motivation

- **Metapath**: an ordered sequence of composite relationships connecting distinct or identical node types.
- For example:
  - P_O: $Sender \xrightarrow{p1\_sends(O)} Message$
  - P_H: $Sender \xrightarrow{s\_has\_domain\_of(H)} Domain$
  - P_{OF} : $Sender \xrightarrow{p1\_sends(O)} Message \xrightarrow{is\_sent\_from(F)} IP$
- P_{OF} is intuitively more closely associated with P_O than P_H due to **greater overlap** in node types and relationships.
- This overlap can be conceptualized as a **parent-child relationship**, where the parent metapath serves as a prefix to its child metapaths.
  - For example, P_O is the parent of P_{OF}.
- Consequently, these parent-child relationships naturally form **a tree hierarchy among the metapaths** (*semantic tree*).

HetTree

Offline Aggregation of Features and Labels

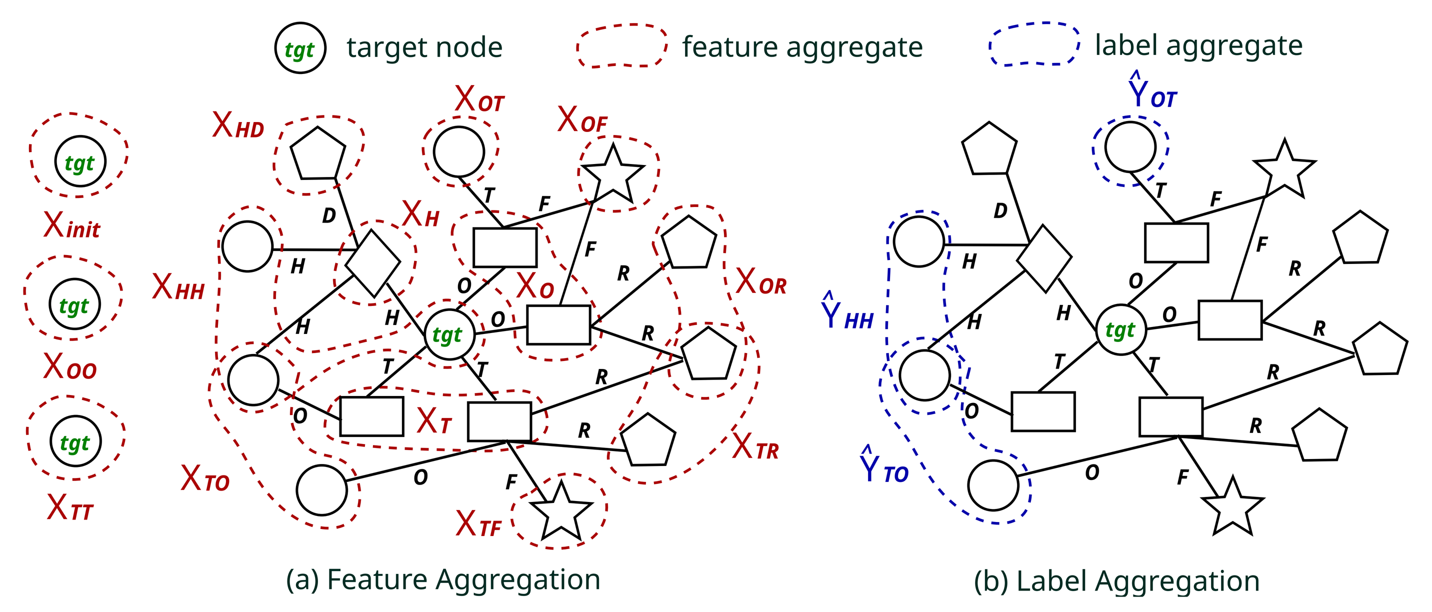

Figure 2. (a) The offline process of feature aggregation. The center node is the target $Sender$ node and features are aggregated for all metapaths $\mathcal{P}^k$ up to hop k, where $k = 2$ in this example. (b) The offline process of label aggregation on partially observed labels in the training set.

Metapath Feature Transformation

HetTree then transforms the aggregated features and labels for metapaths to the same latent space. It **matches and concatenates** (||) **the aggregated features and labels of the same metapath** P. Specifically, for all $P \in \mathcal{P}^k$, we compute the metapath features $\mathcal{M}$ as

$$\mathcal{M} = \{M_P = \begin{cases} MLP(X_P \parallel \hat{Y}_P), & \text{if } P \in \mathcal{P}^k_{O_{tgt}} \\ MLP(X_P), & \text{otherwise} \end{cases}\}. \quad (1)$$

Semantic Tree Aggregation: Subtree Attention

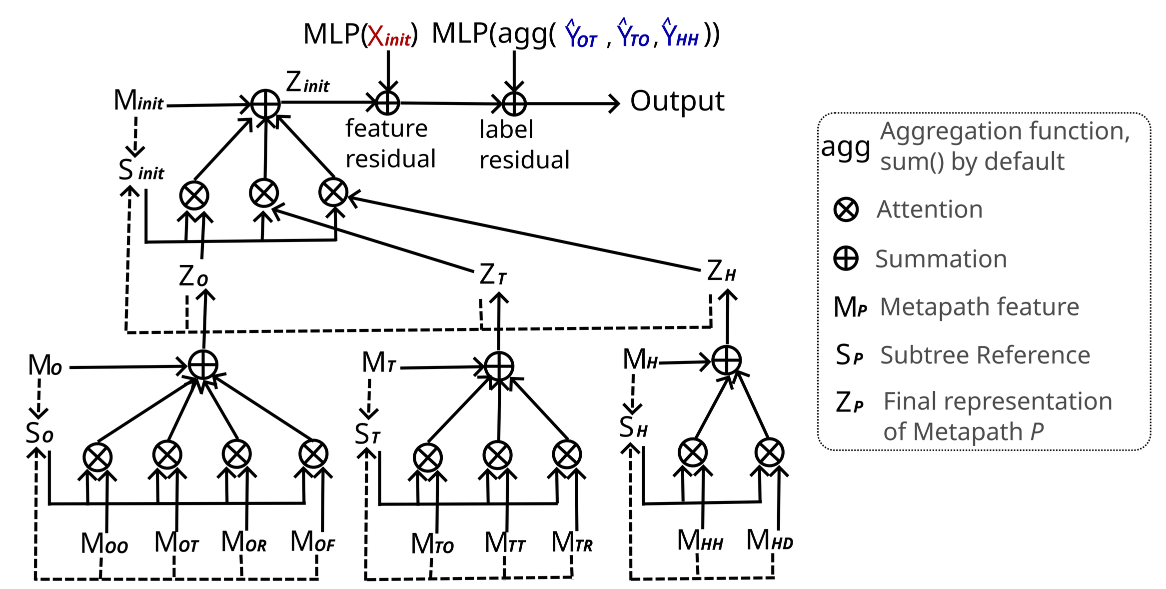

Figure 3. HetTree proposes a novel subtree attention to encode both the parent and children representation and uses it to emphasize the hierarchical correlation among metapaths.

Experiments

We conduct **extensive experiments** on five open graph datasets as well as a real-world commercial email dataset (more details in the paper). The results demonstrate that HetTree can **outperform the state-of-the-art architectures** on all datasets with **low computation and memory overhead**.

Performance on HGB benchmark

	DBLP		IMDB		ACM	
	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1
RGCN	91.52±0.50	92.07±0.50	58.85±0.26	62.05±0.15	91.55±0.74	91.41±0.75
HAN	91.67±0.49	92.05±0.62	57.74±0.96	64.63±0.58	90.89±0.43	90.79±0.43
HetGNN	91.76±0.43	92.33±0.41	48.25±0.67	51.16±0.65	85.91±0.25	86.05±0.25
MAGNN	93.28±0.51	93.76±0.45	56.49±3.20	64.67±1.67	90.88±0.64	90.77±0.65
HGT	93.01±0.23	93.49±0.25	63.00±1.19	67.20±0.57	91.12±0.76	91.00±0.76
HGB	94.01±0.24	94.46±0.22	63.53±1.36	67.36±0.57	93.42±0.44	93.35±0.45
SeHGNN	95.06±0.17	95.42±0.17	67.11±0.25	69.17±0.43	94.05±0.35	93.98±0.36
HetTree	95.34±0.17	95.64±0.15	68.43±0.31	70.92±0.29	94.26±0.20	94.19±0.20

Table 1. Experimental Results of HetTree and baselines on the HGB benchmark.

Epoch Time And Memory Usage

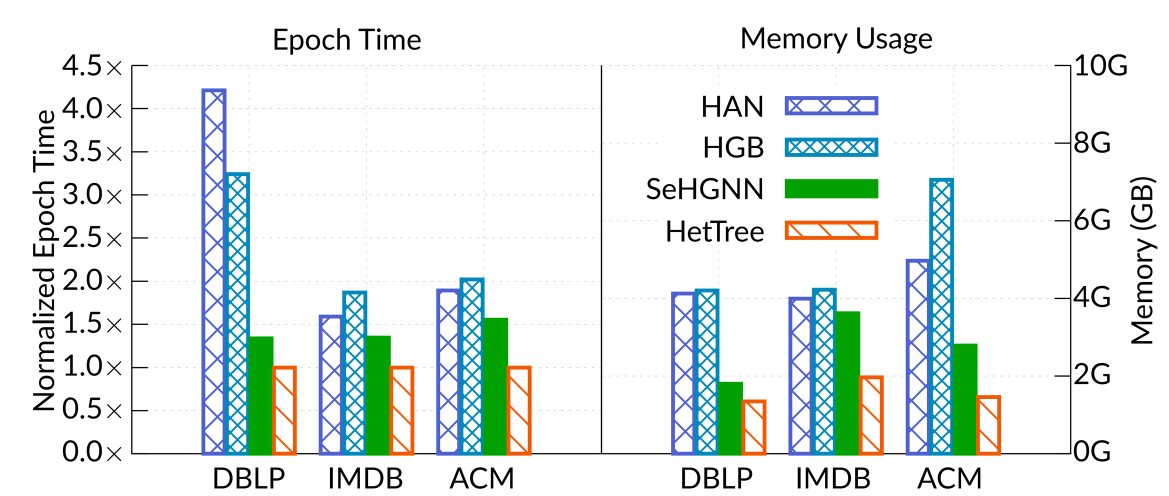

Figure 4. Epoch time and memory usage on HGB datasets.

Takeaways

- Existing HGNNs ignore **a tree hierarchy among metapaths**, which is naturally constituted by different node types and relation types.
- HetTree builds a **semantic tree** structure to capture the hierarchy and proposes a **subtree attention** mechanism to encode the semantic tree.
- A **future direction** is to generalize the semantic tree structure to HGNNs with multi-layer aggregation.

Contact: mingyu.guan@gatech.edu