BENCHTEMP: A General Benchmark for Evaluating Temporal Graph Neural Networks

Authors' Response to Reviewer J4Uw

Opportunities For Improvement:

W1. The novelty and contributions of this work are somewhat limited. No new evaluation tasks or datasets are developed in this work. The evaluation tasks and settings, including the transductive and inductive settings, have been widely used by previous works. Also, the datasets included in the benchmark were created by previous works.
W2. The experiments section focuses a lot on presenting and discussing individual methods' performance. It'd be better to provide summaries of the state-of-the-art results and the limitations of existing methods, and in light of that, discuss future directions of TGNN research.
W3. This paper defines a temporal graph to be a sequence of temporal "user-item" interactions. However, this is a limited form of a temporal graph as this definition covers only a particular type of bipartite graphs with two types of nodes. I think using a more general definition without such conditions (in both the writing and the code) would be more suitable for a general TGNN benchmark.
W4. Datasets are not that large for efficiency evaluation. Most graphs used in the benchmark are not that large. The GPU memory usage for these graphs are mostly 1-3 GB. Larger temporal graphs would be more desirable for evaluating model efficiency. Constructing synthetic temporal graphs with increasing sizes could facilitate more systematic evaluations of TGNN models' efficiency.
W5. Node reindexing described in Figure 3 is confusing. In the homogeneous graph, why do two different nodes have the same id? For example, in the rightmost graph in Fig 3, there are a user with id 2 and an item with id 2. In general, nodes should have different ids as they are separate entities.

General Response:

We appreciate your great feedback! We have presented new datasets with up to several million edges and nodes. We have carefully through your comments and added *six* datasets (eBay-Small, eBay-Large, Taobal-Large, DGraphFin, YouTubeReddit-Small, YouTubeReddit-Large), including *four* **large-scale** datasets (eBay-Large, Taobao-Large, DGraphFin, YouTubeReddit-Large). We have reported the corresponding experiments and detailed discussions in the updated paper. The eBay datasets are a collection of the user transactions on **eBay's e-commerce platform**. We thank our industrial collaborator for sharing their datasets in our research. Considering user privacy and security, eBay datasets could only be shared among collaborators. Any researchers who are interested in the eBay datasets, please email our team (jonnyhuanghnu@gmail.com). For easy access, all datasets have been hosted on the open-source platform zenodo (`https://zenodo.org/`) with a Digital Object Identifier (DOI) 10.5281/zenodo.8267846 (`https://zenodo.org/record/8267846`).

We have conducted extensive experiments on the six newly added temporal graph datasets, including the dynamic link prediction task and dynamic node classification task, diverse workloads (transductive, inductive, inductive New-Old, and inductive New-New scenarios). We have added **the inference**

time metric to evaluate the efficiency of methods. We have added ***Average Rank*** metric shown in Table 2 and Table 4 for ranking model performances on the newly added large-scale datasets for evaluating TGNN models.

We have open sourced the codes of preprocessing large-scale datasets at `https://github.com/qianghuangwhu/benchtemp/tree/master/preprocess` with MIT license.

We have updated the manuscript in section 4.4 to provide summaries of the state-of-the art results and the limitations of existing methods, and the future directions of TGNN research.

We have updated a general definition for temporal graph in section 3.1. A temporal graph can be represented as an ordered sequence of temporal interactions. Each interaction $I_r = (u_r, i_r, t_r, e_r)$ happens at time t_r between the source node u_r and the destination node i_r with edge feature e_r. Considering many real world applications and large-scale temporal graphs,

We have changed the rightmost graph in Figure 3. In the homogeneous graph shown in Fig 3(b), the user with id 2 and the item with id 2 are the same node.

We provide our response to each individual comment below:

Comment 1

W1. The novelty and contributions of this work are somewhat limited. No new evaluation tasks or datasets are developed in this work. The evaluation tasks and settings, including the transductive and inductive settings, have been widely used by previous works. Also, the datasets included in the benchmark were created by previous works.

Response:

We thank the reviewer for the suggestions! Indeed, the evaluation settings have been widely used by previous works, which are also adopted by SOTA methods. Our purpose is not to introduce new settings; instead, we aim at comparing different TGNN models on the same ground.

We have included new datasets with up to several million edges and nodes. We have added *six* datasets (eBay-Small, eBay-Large, Taobal-Large, DGraphFin, YouTubeReddit-Small, YouTubeReddit-Large), including *four* **large-scale** datasets (eBay-Large, Taobao-Large, DGraphFin, YouTubeReddit-Large). The statistics of the new datasets are shown in Table 1. For easy access, all datasets have been hosted on the open-source platform zenodo (`https://zenodo.org/`) with a Digital Object Identifier (DOI) 10.5281/zenodo.8267846 (`https://zenodo.org/record/8267846`). In our paper, we present BenchTemp, a general benchmark for evaluating temporal graph neural network (TGNN) models over a wide range of tasks and settings. We extensively compare representative TGNN models on the benchmark datasets, regarding different tasks, settings, metrics, and especially model efficiency - **inference time.**

- **eBay-Small** is a subset of the eBay-Large dataset. We sample 38,427 nodes and 384,677 edges from eBay-Large graph according to edge timestamps.

Table 1: Dataset statistics of the new datasets.

	Domain	*# Nodes*	*# Edges*
eBay-Small	E-commerce	38,427	384,677
YouTubeReddit-Small [1]	Social	264,443	297,732
eBay-Large	E-commerce	1,333,594	1,119,454
DGraphFin [2]	E-commerce	3,700,550	4,300,999
Youtube-Reddit-Large [1]	Social	5,724,111	4,228,523
Taobao-Large [3, 4]	E-commerce	1,630,453	5,008,745

- **YouTubeReddit-Small** is a collection of massive visual contents on YouTube and long-term community activity on Reddit. This dataset covers a **3**-month period from January to March 2020. Each row in the dataset represents a YouTube video v_i being shared in a subreddit s_j by some user u_k at time t [1]. Nodes are YouTube videos and subreddits, edges are the users' interactions between videos and subreddits. This dynamic graph has 264,443 nodes and 297,732 edges.
- **eBay-Large** is a million-scale dataset consisting of 1.3 million nodes and 1.1 million edges, which comprises the selected transaction records from the eBay e-commerce platform over a two-month period. eBay-Large is modeled as a user-item graph, where items are heterogeneous entities which include information such as phone numbers, addresses, and email addresses associated with a transaction. We selecte one month of transactions as seed nodes and then expand each seed node two hops back in time to enrich the topology while maintaining consistency in the distribution of seed nodes.
- **DGraphFin** is a collection of large-scale dynamic graph datasets, consisting of interactive objects, events and labels that evolve with time.It is a directed, unweighted dynamic graph consisting of millions of nodes and edges, representing a realistic user-to-user social network in financial industry. Nodes are users, and an edge from one user to another means that the user regards the other user as the emergency contact person [2].
- **Youtube-Reddit-Large** dataset covers **54** months of YouTube video propagation history from January 2018 to June 2022 [1]. This dataset has 5,724,111 nodes and 4,228,523 edges.
- **Taobao-Large** is a collection of the Taobao user behavior dataset intercepted based on the period 8:00 to 18:00 on 26 November 2017 [4]. Nodes are users and items, and edges are behaviors between users and items, such as favor, click, purchase, and add an item to shopping cart. This public dataset has 1,630,453 nodes and 5,008,74 user-item interaction edges.

A Experiments

We have conducted extensive experiments on the six newly added temporal graph datasets, including the dynamic link prediction task and dynamic node classification task, diverse workloads (transductive, inductive, inductive New-Old, and inductive New-New scenarios). We have added **the inference time** metric to evaluate the efficiency of methods.

The experimental setup is the same as in the paper.

A.1 Link Prediction Task

We run the link prediction task on 7 TGNN models and the new datasets under different settings (Transductive, Inductive, Inductive New-Old, and Inductive New-New).

The experimental results on large-scale datasets may be more convincing. Furthermore, we have added ***Average Rank*** metric shown in Table 2 and Table 4 for ranking model performances on the newly added large-scale datasets for evaluating TGNN models.

The AUC and AP results for each new datasets are shown in Table 2 and Table 3, respectively. For the four large-scale datasets (eBay-Large, Taobao-Large, DGraphFin, YouTubeReddit-Large), we observe the similar results as in the paper. Specifically, NAT and NeurTW achieve the top-2 performance on almost all datasets under transductive and inductive settings.

A.2 Node Classification Task

The eBay-Small and eBay-Large datasets have node labels, so we conduct dynamic node classification experiments on both the eBay-Small and eBay-Large datasets. The AUC results are shown in Table 4. We can observe the similar results as in the paper. NeurTW achieves the best performance on both eBay-Small and eBay-Large datasets. NAT performs poorly on the node classification task.

Table 2: ROC AUC results of new datasets on the *dynamic link prediction task*. The best and second-best results are highlighted as **bold red** and underlined blue. **Average Rank** are computed by the experimental results of models on four large-scale datasets (eBay-Large, Taobao-Large, DGraphFin, YouTubeReddit-Large). We do not highlight the second-best if the gap is > 0.05 compared with the best result.

Dataset \ Model	JODIE	DyRep	TGN	TGAT	CAWN	NeurTW	NAT
				Transductive			
eBay-Small	0.9946 ± 0.0002	0.9941 ± 0.0006	0.9984 ± 0.0003	0.9838 ± 0.0006	0.9985 ± 0.0	**0.9991 ± 0.0**	0.9978 ± 0.0003
YouTubeReddit-Small	0.8519 ± 0.0007	0.8499 ± 0.0012	0.8432 ± 0.0032	0.8441 ± 0.0014	0.7586 ± 0.0031	**0.9003 ± 0.0031**	0.8259 ± 0.005
eBay-Large	0.9614 ± 0.0	0.9619 ± 0.0001	0.9642 ± 0.0003	0.5311 ± 0.0003	0.9442 ± 0.0003	0.9608 ± 0.0	**0.9658 ± 0.0002**
DGraphFin	0.8165 ± 0.0024	0.8171 ± 0.0016	**0.8683 ± 0.0023**	0.6112 ± 0.0165	0.5466 ± 0.0103	0.8611 ± 0.0035	0.8258 ± 0.0001
Youtube-Reddit-Large	0.8532 ± 0.0003	0.8529 ± 0.0006	0.8458 ± 0.0025	0.8536 ± 0.0026	0.7466 ± 0.0012	**0.916 ± 0.0025**	0.8605 ± 0.0009
Taobao-Large	0.7726 ± 0.0005	0.7724 ± 0.001	0.8464 ± 0.0008	0.5567 ± 0.0047	0.7771 ± 0.0068	**0.859 ± 0.0091**	0.8188 ± 0.001
Average Rank	4.5	4.5	2.75	5.75	6	2.25	2.25
				Inductive			
eBay-Small	0.9696 ± 0.0007	0.9674 ± 0.0018	0.9913 ± 0.0004	0.9698 ± 0.0006	0.9964 ± 0.0001	0.9982 ± 0.0	**0.9998 ± 0.0001**
YouTubeReddit-Small	0.7582 ± 0.0003	0.7545 ± 0.0009	0.7276 ± 0.0033	0.7436 ± 0.0006	0.7533 ± 0.0016	0.8978 ± 0.0032	**0.9876 ± 0.0049**
eBay-Large	0.7536 ± 0.0014	0.7515 ± 0.0006	0.7657 ± 0.0026	0.5224 ± 0.0003	0.9459 ± 0.0001	0.9608 ± 0.0	**0.9999 ± 0.0001**
DGraphFin	0.6884 ± 0.0051	0.6876 ± 0.001	0.6439 ± 0.0089	0.5677 ± 0.0184	0.5479 ± 0.009	**0.8635 ± 0.0021**	0.7955 ± 0.0201
Youtube-Reddit-Large	0.7539 ± 0.0005	0.7554 ± 0.0003	0.7243 ± 0.0016	0.7501 ± 0.0019	0.7327 ± 0.0016	0.9128 ± 0.0031	**0.9863 ± 0.006**
Taobao-Large	0.7075 ± 0.0009	0.7042 ± 0.0006	0.6812 ± 0.0032	0.5222 ± 0.0041	0.7787 ± 0.0103	0.869 ± 0.010	**0.9933 ± 0.0008**
Average Rank	4	4.5	5.5	6.25	4.75	1.75	1.25
				Inductive New-Old			
eBay-Small	0.9862 ± 0.0003	0.9836 ± 0.0016	0.9947 ± 0.0009	0.9712 ± 0.002	0.9985 ± 0.0	0.9988 ± 0.0	**0.9999 ± 0.0**
YouTubeReddit-Small	0.7695 ± 0.001	0.7655 ± 0.0018	0.7396 ± 0.0034	0.7242 ± 0.0004	0.7573 ± 0.0022	0.922 ± 0.0002	**0.9967 ± 0.0014**
eBay-Large	0.6109 ± 0.0244	0.5906 ± 0.0087	0.8134 ± 0.0105	0.6363 ± 0.0605	0.9569 ± 0.0007	0.8973 ± 0.0	**1.0 ± 0.0**
DGraphFin	0.5768 ± 0.0071	0.5735 ± 0.0007	0.5564 ± 0.0021	0.5742 ± 0.013	0.5646 ± 0.0244	0.7702 ± 0.0043	**0.8693 ± 0.0066**
Youtube-Reddit-Large	0.7844 ± 0.0015	0.7894 ± 0.0017	0.7623 ± 0.0031	0.7457 ± 0.0062	0.7511 ± 0.0022	0.9356 ± 0.0004	**0.9958 ± 0.0025**
Taobao-Large	0.7023 ± 0.0015	0.6953 ± 0.0022	0.6771 ± 0.0055	0.5104 ± 0.0106	0.7674 ± 0.005	0.8458 ± 0.0043	**0.9965 ± 0.0005**
Average Rank	4.25	5	5.5	5.75	4.25	2.25	1
				Inductive New-New			
eBay-Small	0.9388 ± 0.0009	0.9366 ± 0.0037	0.9838 ± 0.0007	0.9556 ± 0.0007	0.9937 ± 0.0	0.9975 ± 0.0	**0.9997 ± 0.0004**
YouTubeReddit-Small	0.7436 ± 0.0015	0.7436 ± 0.0018	0.7265 ± 0.0055	0.749 ± 0.0011	0.7479 ± 0.004	0.864 ± 0.0071	**0.9868 ± 0.0049**
eBay-Large	0.7526 ± 0.0013	0.7500 ± 0.0005	0.7639 ± 0.0027	0.5196 ± 0.0002	0.9542 ± 0.0003	0.9615 ± 0.0	**0.9999 ± 0.0001**
DGraphFin	0.7307 ± 0.0007	0.7323 ± 0.0002	0.6843 ± 0.0131	0.5649 ± 0.0248	0.5417 ± 0.0099	**0.9051 ± 0.0028**	0.7584 ± 0.0323
Youtube-Reddit-Large	0.6932 ± 0.0026	0.7022 ± 0.0007	0.6703 ± 0.0024	0.7269 ± 0.0	0.6942 ± 0.0028	0.8716 ± 0.0077	**0.9796 ± 0.0103**
Taobao-Large	0.7243 ± 0.0001	0.7247 ± 0.0001	0.6885 ± 0.0024	0.5256 ± 0.0054	0.7922 ± 0.0118	0.8906 ± 0.0088	**0.9969 ± 0.0002**
Average Rank	5	4.25	5.5	5.75	4.5	1.75	1.25
	JODIE	DyRep	TGN	TGAT	CAWN	NeurTW	NAT
Total Rank	4.44	4.56	4.81	5.88	4.88	2.00	**1.44**

A.3 Efficiency

Considering many real world applications and , we add **the inference time** metric to evaluate the efficiency of models. The inference time comparison per 100,000 edges is shown in Figure 1. According to the figure, we can observe the similar model efficiency results as in the paper. In terms of the inference time, JODIE, DyRep, TGN and TGAT are faster, while CAWN and NeurTW are much slower. NAT is relatively faster than temporal walk-based methods through caching and parallelism optimizations, *achieving a good trade-off between model quality and efficiency*.

Table 3: AP results of new datasets on the *dynamic link prediction task*. The best and second-best results are highlighted as **bold red** and underlined blue. We do not highlight the second-best if the gap is > 0.05 compared with the best result.

Dataset \ Model	JODIE	DyRep	TGN	TGAT	CAWN	NeurTW	NAT
Transductive							
eBay-Small	0.9938 ± 0.0004	0.9936 ± 0.0006	0.9983 ± 0.0003	0.9819 ± 0.0009	0.9981 ± 0.0	**0.9991 ± 0.0**	0.9975 ± 0.0002
YouTubeReddit-Small	0.8612 ± 0.0009	0.8594 ± 0.0012	0.8421 ± 0.0041	0.8515 ± 0.0012	0.7625 ± 0.0042	**0.9112 ± 0.0021**	0.8325 ± 0.0068
eBay-Large	0.9318 ± 0.0002	0.9322 ± 0.0002	0.9357 ± 0.0006	0.5239 ± 0.0002	0.9144 ± 0.0004	0.9307 ± 0.0	**0.9398 ± 0.0004**
DGraphFin	0.7705 ± 0.0009	0.7705 ± 0.0024	0.8571 ± 0.0009	0.6441 ± 0.0123	0.5431 ± 0.0095	**0.8637 ± 0.0014**	0.7956 ± 0.0012
Youtube-Reddit-Large	0.8622 ± 0.0007	0.8632 ± 0.0004	0.8476 ± 0.0022	0.8591 ± 0.0026	0.7475 ± 0.0017	**0.9222 ± 0.0013**	0.8628 ± 0.0015
Taobao-Large	0.7164 ± 0.0003	0.7142 ± 0.0008	0.844 ± 0.0011	0.5761 ± 0.0023	0.7616 ± 0.0069	**0.8568 ± 0.016**	0.7904 ± 0.0008
Inductive							
eBay-Small	0.9638 ± 0.0007	0.9619 ± 0.0017	0.9898 ± 0.0005	0.9675 ± 0.0007	0.9953 ± 0.0002	0.9982 ± 0.0	**0.9998 ± 0.0001**
YouTubeReddit-Small	0.7866 ± 0.0007	0.7833 ± 0.0009	0.7387 ± 0.0069	0.7551 ± 0.0002	0.7568 ± 0.0031	0.9086 ± 0.0022	**0.9872 ± 0.0056**
eBay-Large	0.6989 ± 0.0018	0.6973 ± 0.0007	0.7096 ± 0.0030	0.518 ± 0.0002	0.9174 ± 0.0001	0.9308 ± 0.0	**0.9999 ± 0.0001**
DGraphFin	0.6563 ± 0.002	0.6567 ± 0.0009	0.624 ± 0.006	0.5866 ± 0.0123	0.5428 ± 0.0082	**0.8626 ± 0.0012**	0.7053 ± 0.0185
Youtube-Reddit-Large	0.7796 ± 0.0009	0.7818 ± 0.0009	0.73 ± 0.0029	0.7587 ± 0.0025	0.7353 ± 0.0022	0.9192 ± 0.0022	**0.9849 ± 0.0071**
Taobao-Large	0.6763 ± 0.0011	0.6746 ± 0.0011	0.6664 ± 0.0012	0.5315 ± 0.0027	0.7533 ± 0.011	0.8596 ± 0.0205	**0.9941 ± 0.0007**
Inductive New-Old							
eBay-Small	0.9849 ± 0.0007	0.9836 ± 0.0013	0.9931 ± 0.0008	0.9682 ± 0.0028	0.9985 ± 0.0001	0.999 ± 0.0	**0.9999 ± 0.0**
YouTubeReddit-Small	0.7963 ± 0.0013	0.7937 ± 0.0014	0.729 ± 0.0086	0.7296 ± 0.0013	0.762 ± 0.0041	0.9244 ± 0.0015	**0.9966 ± 0.0016**
eBay-Large	0.5670 ± 0.0186	0.5870 ± 0.0074	0.8024 ± 0.0060	0.6504 ± 0.0385	0.9592 ± 0.0008	0.8458 ± 0.0	**1.0 ± 0.0**
DGraphFin	0.6005 ± 0.0048	0.5872 ± 0.0059	0.5753 ± 0.0062	0.5927 ± 0.0058	0.5669 ± 0.0269	0.7572 ± 0.0025	**0.8184 ± 0.0088**
Youtube-Reddit-Large	0.808 ± 0.0014	0.8142 ± 0.0019	0.7472 ± 0.0043	0.7526 ± 0.0097	0.7553 ± 0.0025	0.9368 ± 0.0009	**0.9953 ± 0.0028**
Taobao-Large	0.7009 ± 0.0013	0.698 ± 0.0014	0.6879 ± 0.0008	0.5254 ± 0.0074	0.7597 ± 0.0053	0.8459 ± 0.0103	**0.9969 ± 0.0004**
Inductive New-New							
eBay-Small	0.923 ± 0.001	0.9226 ± 0.0024	0.98 ± 0.0007	0.9505 ± 0.0009	0.991 ± 0.0001	0.9973 ± 0.0	**0.9997 ± 0.0004**
YouTubeReddit-Small	0.7578 ± 0.0015	0.7582 ± 0.0021	0.7564 ± 0.0043	0.7718 ± 0.0023	0.7498 ± 0.004	0.8868 ± 0.0034	**0.9861 ± 0.0063**
eBay-Large	0.6976 ± 0.0016	0.6957 ± 0.0007	0.7078 ± 0.0031	0.5154 ± 0.0001	0.93 ± 0.0003	0.9318 ± 0.0	**0.9999 ± 0.0001**
DGraphFin	0.6802 ± 0.0005	0.6811 ± 0.0002	0.6526 ± 0.0098	0.5831 ± 0.0184	0.5379 ± 0.0071	**0.8977 ± 0.0014**	0.6529 ± 0.0249
Youtube-Reddit-Large	0.7038 ± 0.0024	0.7115 ± 0.0007	0.6979 ± 0.002	0.7414 ± 0.0012	0.6965 ± 0.004	0.8848 ± 0.0023	**0.9761 ± 0.0134**
Taobao-Large	0.6738 ± 0.0005	0.6742 ± 0.0005	0.6611 ± 0.0011	0.53 ± 0.0023	0.7521 ± 0.0127	0.8738 ± 0.0145	**0.9973 ± 0.0001**

Table 4: ROC AUC results for the *dynamic node classification task* on the eBay datasets. The top-2 results are highlighted as **bold red** and underlined blue.

Dataset \ Model	JODIE	DyRep	TGN	TGAT	CAWN	NeurTW	NAT
eBay-Small	0.9274 ± 0.0017	0.8677 ± 0.0356	0.913 ± 0.0025	0.9342 ± 0.0002	0.9305 ± 0.0001	**0.9529 ± 0.0002**	0.6797 ± 0.0115
eBay-Large	0.7244 ± 0.0002	0.7246 ± 0.0	0.6586 ± 0.0129	0.672 ± 0.0016	0.7710 ± 0.0002	**0.7859 ± 0.0**	0.5304 ± 0.0011
Average Rank	4	4.5	5.5	3.5	2.5	**1**	7

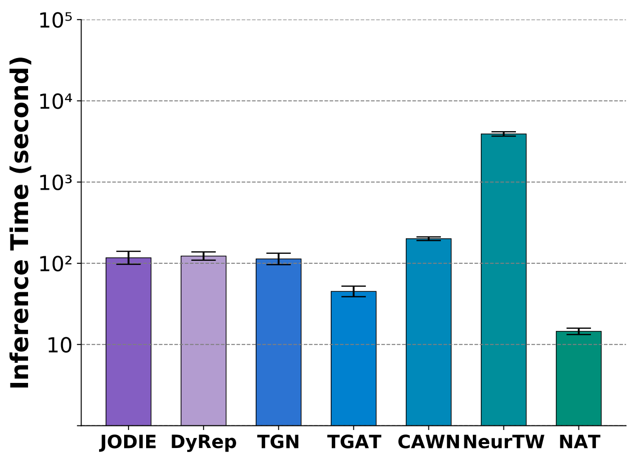

Figure 1: Inference time comparison per 100,000 edges.

Comment 2

W2. The experiments section focuses a lot on presenting and discussing individual methods' performance. It'd be better to provide summaries of state-of-the-art results and the limitations of existing methods, and in light of that, discuss future directions of TGNN research.

Response:

We appreciate this suggestion! We have updated the paper (`https://openreview.net/pdf?id=rnZm2vQq31`) in section 4.4 to provide the summaries of the state-of-the-art results and the limitations of existing methods, and the future directions of TGNN research.

> **Comment 3**
>
> **W3.** This paper defines a temporal graph to be a sequence of temporal "user-item" interactions. However, this is a limited form of a temporal graph as this definition covers only a particular type of bipartite graphs with two types of nodes. I think using a more general definition without such conditions (in both the writing and the code) would be more suitable for a general TGNN benchmark.

Response:

Thanks for your comment! We have updated a general definition for temporal graph in section 3.1. A temporal graph can be represented as an ordered sequence of temporal interactions. The r-th interaction $I_r = (u_r, i_r, t_r, e_r)$ happens at time t_r between the source node u_r and the destination node i_r with edge feature e_r.

> **Comment 4**
>
> **W4.** Datasets are not that large for efficiency evaluation. Most graphs used in the benchmark are not that large. The GPU memory usage for these graphs are mostly 1-3 GB. Larger temporal graphs would be more desirable for evaluating model efficiency. Constructing synthetic temporal graphs with increasing sizes could facilitate more systematic evaluations of TGNN models' efficiency.

Response:

Thanks for this valuable comment!

We have included new datasets with up to several million edges and nodes. We have added *six* datasets (eBay-Small, eBay-Large, Taobal-Large, DGraphFin, YouTubeReddit-Small, YouTubeReddit-Large), including ***four*** **large-scale** datasets (eBay-Large, Taobao-Large, DGraphFin, YouTubeReddit-Large). The statistics of the new datasets are shown in Table 1. For easy access, all datasets have been hosted on the open-source platform zenodo (`https://zenodo.org/`) with a Digital Object Identifier (DOI) 10.5281/zenodo.8267846 (`https://zenodo.org/record/8267846`). Furthermore, considering many real world applications and large-scale temporal graphs, we have added **the inference time** metric to evaluate the efficiency of TGNN models. See Section A for details.

> **Comment 5**
>
> **W5.** Node reindexing described in Figure 3 is confusing. In the homogeneous graph, why do two different nodes have the same id? For example, in the rightmost graph in Fig 3, there are a user with id 2 and an item with id 2. In general, nodes should have different ids as they are separate entities.

Response:

We appreciate the suggestion and totally agree. In the homogeneous graph shown in Figure 3(b), the user with id 2 and the item with id 2 are the same node. Therefore, we have updated the Figure 3(b) in the paper by replacing the labels "user" and "item" with "node".

References

[1] Yiqiao Jin, Yeon-Chang Lee, Kartik Sharma, Meng Ye, Karan Sikka, Ajay Divakaran, and Srijan Kumar. Predicting information pathways across online communities. *arXiv preprint arXiv:2306.02259*, 2023.

[2] Xuanwen Huang, Yang Yang, Yang Wang, Chunping Wang, Zhisheng Zhang, Jiarong Xu, Lei Chen, and Michalis Vazirgiannis. Dgraph: A large-scale financial dataset for graph anomaly detection. *Advances in Neural Information Processing Systems*, 35:22765–22777, 2022.

[3] Ming Jin, Yuan-Fang Li, and Shirui Pan. Neural temporal walks: Motif-aware representation learning on continuous-time dynamic graphs. In *Advances in Neural Information Processing Systems*, 2022.

[4] Han Zhu, Xiang Li, Pengye Zhang, Guozheng Li, Jie He, Han Li, and Kun Gai. Learning tree-based deep model for recommender systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1079–1088, 2018.