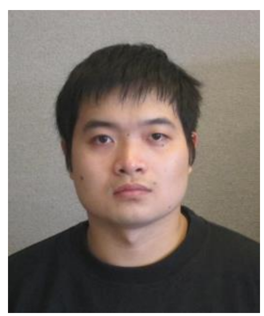
A Data-Driven Graph Generative Model for Temporal Interaction Networks

Presenter: Dawei Zhou

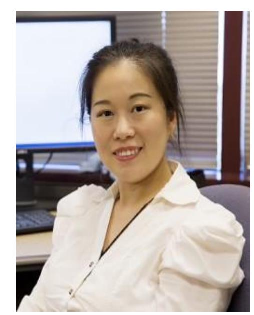
Contact: dzhou21@Illinois.edu

Dawei Zhou
(UIUC)



Lecheng Zheng
(UIUC)

Jiawei Han
(UIUC)



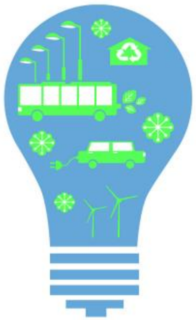
Jingrui He
(UIUC)

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Temporal Networks

E-commerce

Finance



Infrastructure

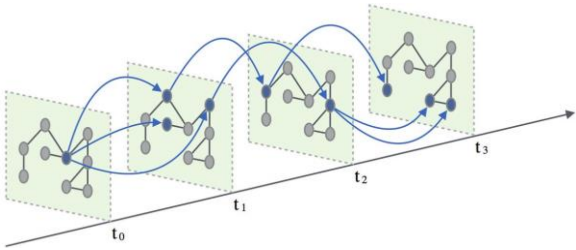
Healthcare

Social Media

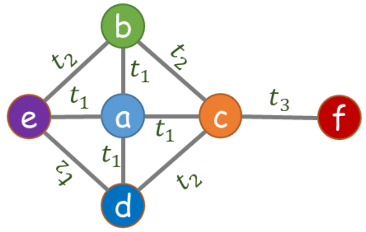
...

• Holme, Petter, and Jari Saramäki. "Temporal networks." Physics reports 519.3 (2012): 97-125.

Temporal Network Representations



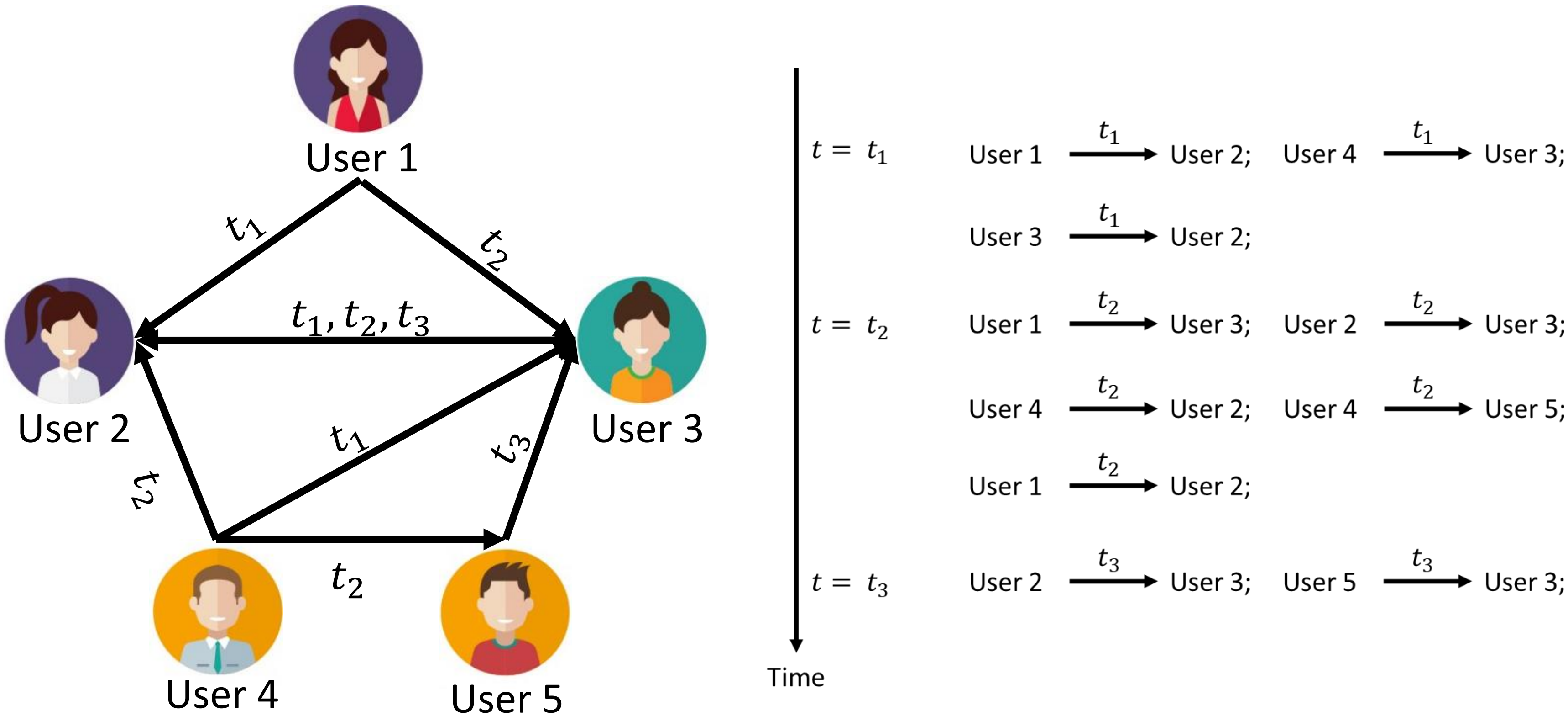
- Time-evolving graphs
  - Aggregate timestamps into a sequence of snapshots.
  - (+): Static graph-based algorithms can be easily applied.
  - (-): Designed for discrete timestamps.
  - (-): Lost fine-grained temporal information during time aggregation.



- Temporal interaction networks
  - Represented as a collection of timestamped edges.
  - (+): Designed for continuous timestamps.
  - (+): Preserve fine-grained dynamics.
  - (-): Traditional graph-based algorithms can not be applied.

- Sharma, Shalini, and Jerry Chou. "A survey of computation techniques on time evolving graphs." International Journal of Big Data Intelligence 7.1 (2020): 1-14.
- Kumar, Srijan, Xikun Zhang, and Jure Leskovec. "Predicting dynamic embedding trajectory in temporal interaction networks." Proceedings of the 25th ACM SIGKDD 2019.

Temporal Interaction Networks

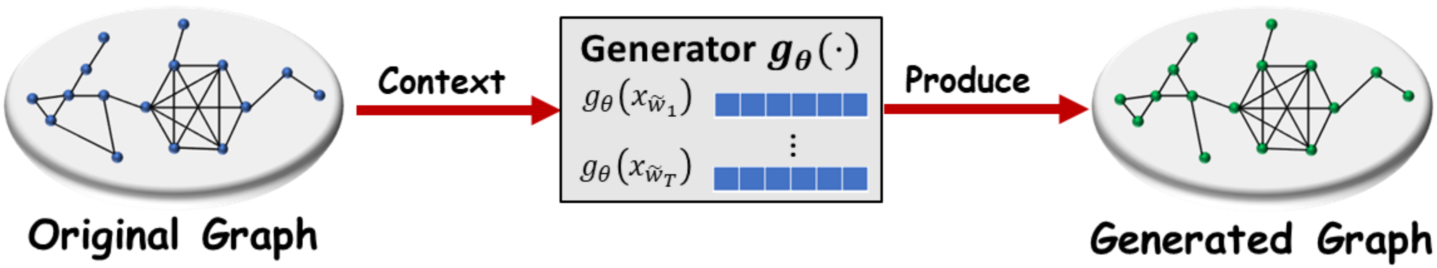


(a) Online Transaction Network

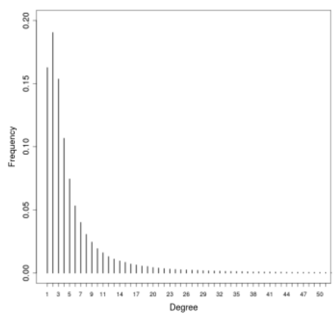
(b) Temporal Interactions

- Kumar, Srijan, Xikun Zhang, and Jure Leskovec. "Predicting dynamic embedding trajectory in temporal interaction networks." Proceedings of the 25th ACM SIGKDD 2019.

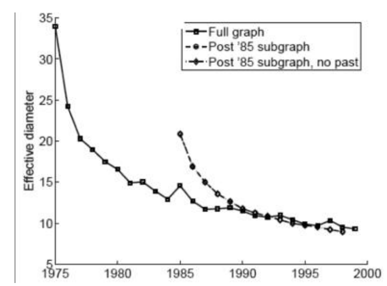
Graph Generative Models



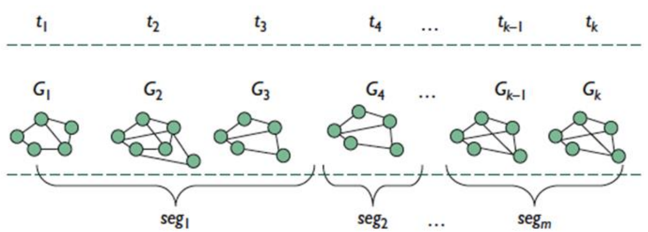
- Reason 1: Discovering structural “laws” in temporal networks



Degree Distribution



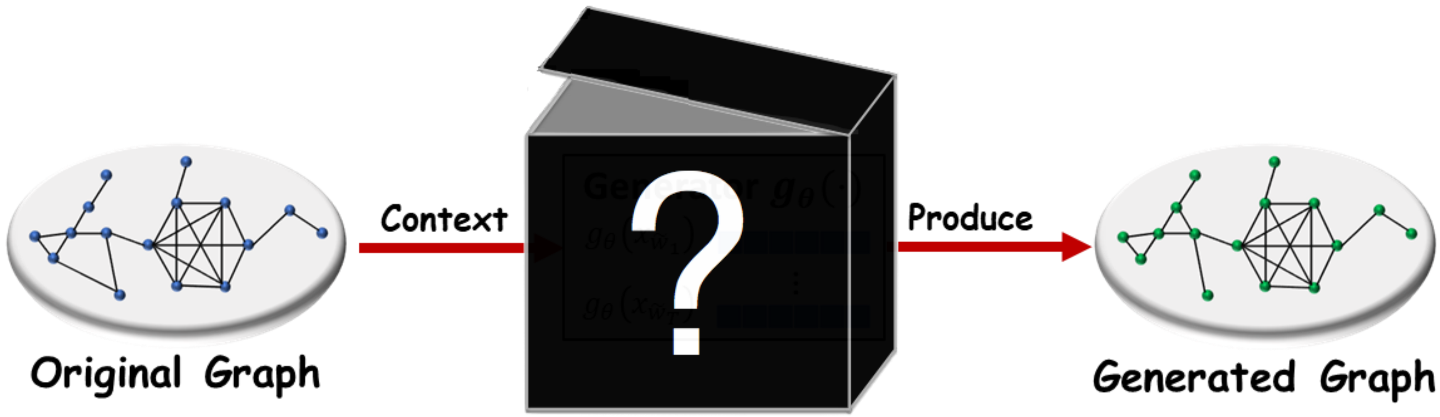
Shrinking Diameter



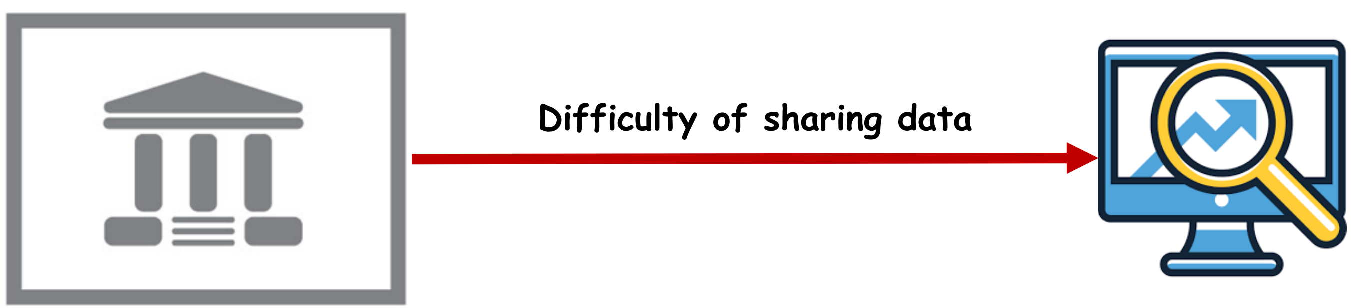
Motif Evolution

- Purohit, Sumit, Lawrence B. Holder, and George Chin. "Temporal graph generation based on a distribution of temporal motifs." Proceedings of the 14th International Workshop on Mining and Learning with Graphs. Vol. 7. 2018.

Graph Generative Models

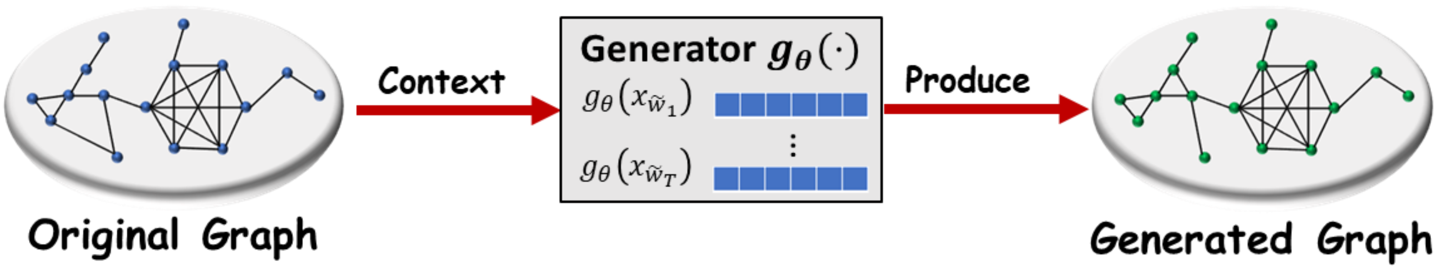


- Reason 2: Data security



- E. W. T. Ngai, Yong Hu, Y. H. Wong, Yijun Chen, Xin Sun: The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature. Decision Support System 50(3): 559-569 (2011)

Graph Generative Models



- Reason 3: Downstream tasks

Data Augmentation

Anomaly Detection

Recommendation

- Jiaxuan You, Rex Ying, Xiang Ren, William L. Hamilton, Jure Leskovec: GraphRNN: Generating Realistic Graphs with Deep Auto-regressive Models. ICML 2018: 5694-5703
- Aleksandar Bojchevski, Oleksandr Shchur, Daniel Zügner, Stephan Günnemann: NetGAN: Generating Graphs via Random Walks. ICML 2018: 609-618

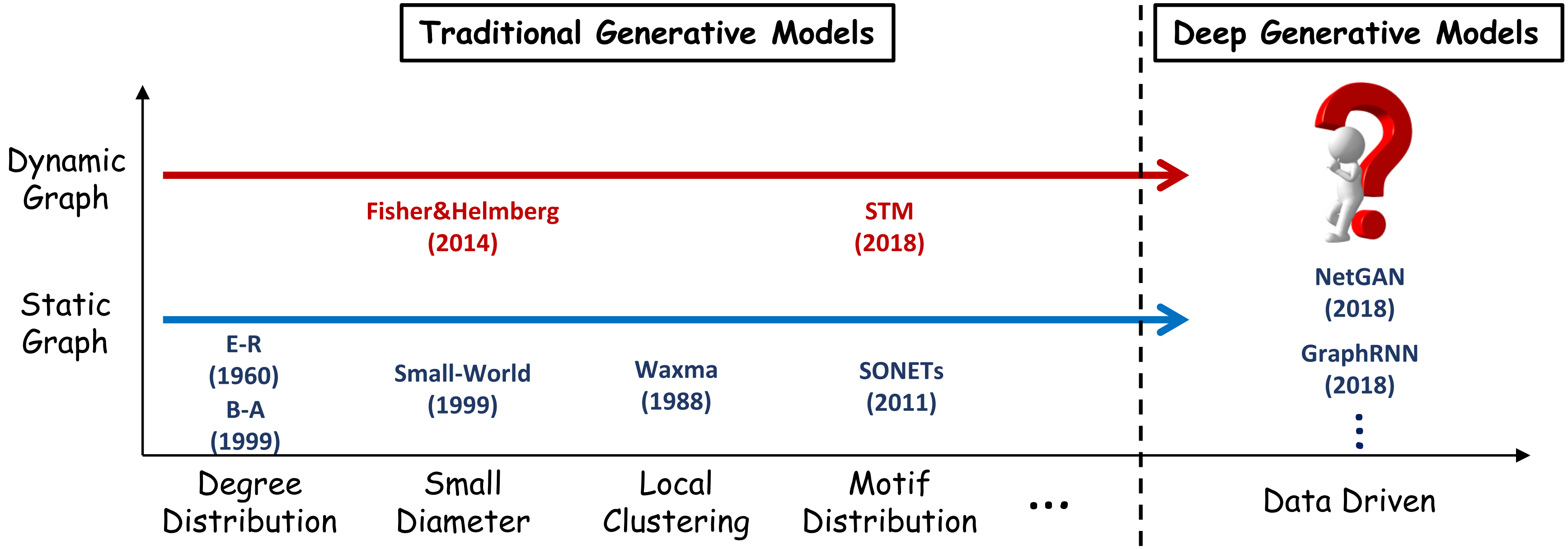
Existing Graph Generative Models

- Traditional Generative Models
  - Rely on structural assumptions.
    - EX: degree distribution, motif distribution, etc.
  - **Pros:**
    - Simple while elegant mathematical properties.
    - Fast.
  - **Cons:**
    - Restrict to one/few structural assumptions.

- Deep Generative Models
  - Trained from the extracted network context information.
    - EX: BFS, random walks, adjacency matrix, etc.
  - **Pros:**
    - Minimum structural assumptions.
    - Superior performance in various metrics.
  - **Cons:**
    - High time complexity.

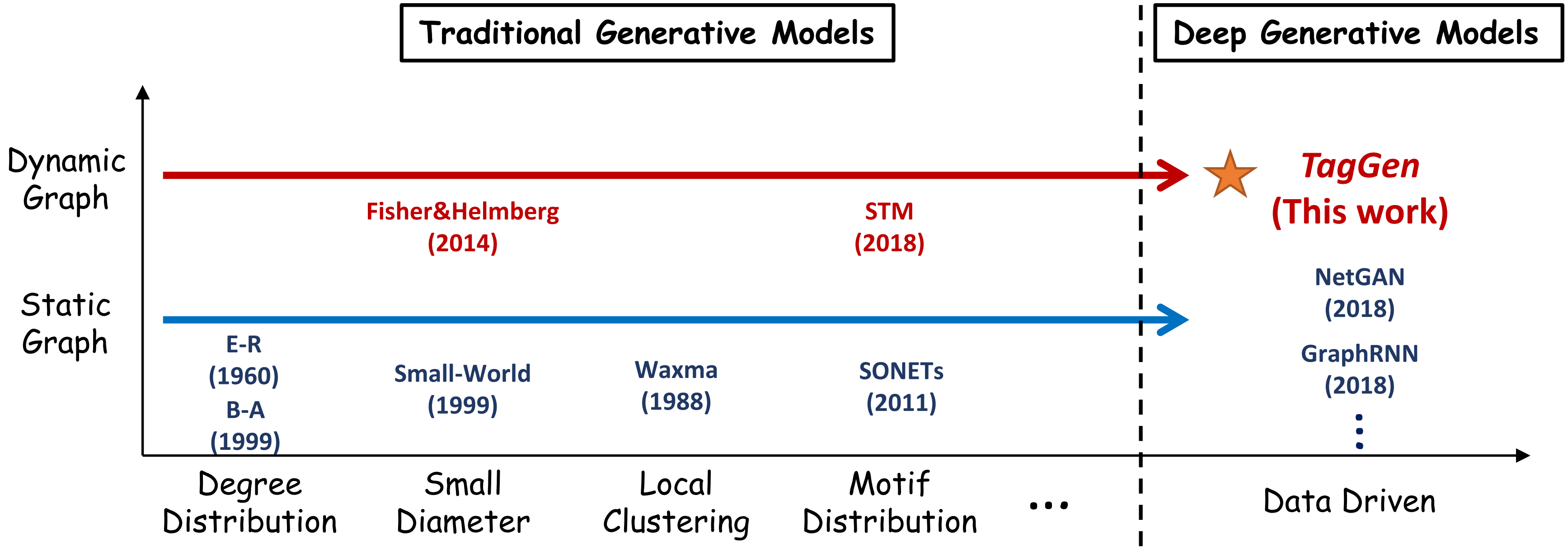
Existing Graph Generative Models

- A two-dimensional conceptual space



Existing Graph Generative Models

- A two-dimensional conceptual space



Challenges

- C1: Can we directly learn from the temporal interaction networks, that are represented in timestamped edges ?
- C2: Can we ensure the generated graphs preserve the structural and temporal characteristics of real graphs?
- C3: What is the impact of our generative model for the downstream applications, such as anomaly detection and recommendation?

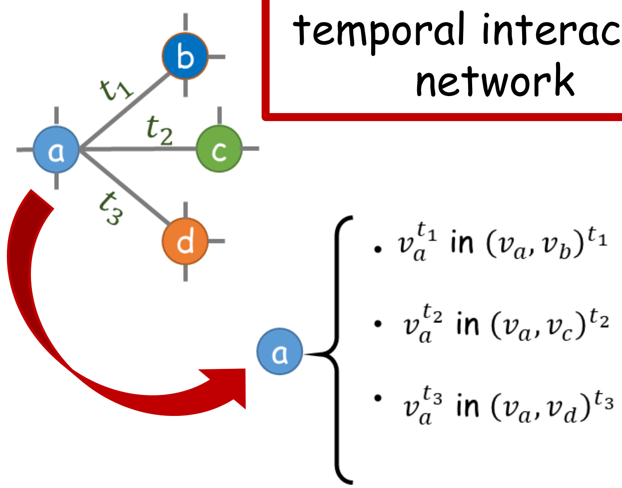
Roadmap

- Motivation
- Problem Definition
- Proposed *TagGen* Framework
- Experiments
- Conclusion

Temporal Node and Temporal Occurrence

- Definition 1. Temporal Node and Temporal Occurrence
  In a temporal interaction network, a node v is associated with a bag of temporal occurrences $v = \{v^{t_1}, v^{t_2}, ...\}$ which instantiate the occurrences of node v at $\{t_1, t_2, ...\}$ in the network.

A miniature of a temporal interaction network

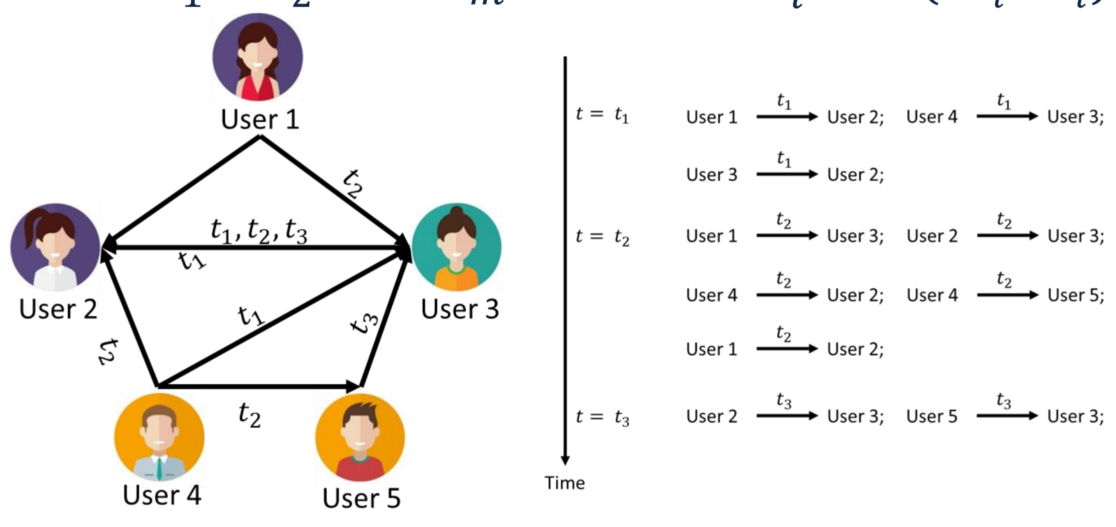


The different occurrence of node v_a that appear at t_1, t_2, t_3.

Temporal Interaction Network

- Definition 2. Temporal Interaction Network

  A temporal interaction network $\tilde{G} = (\tilde{V}, \tilde{E})$ is formed by a collection of nodes $\tilde{V} = \{v_1, v_2, \ldots, v_n\}$ and a series of timestamped edges $\tilde{E} = (e_1^{t_{e_1}}, e_2^{t_{e_2}}, \ldots, e_m^{t_{e_m}})$, where $e_i^{t_{e_i}} = (u_{e_i}, v_{e_i})^{t_{e_i}}$.



(a) Online Transaction Network (b) Temporal Interactions

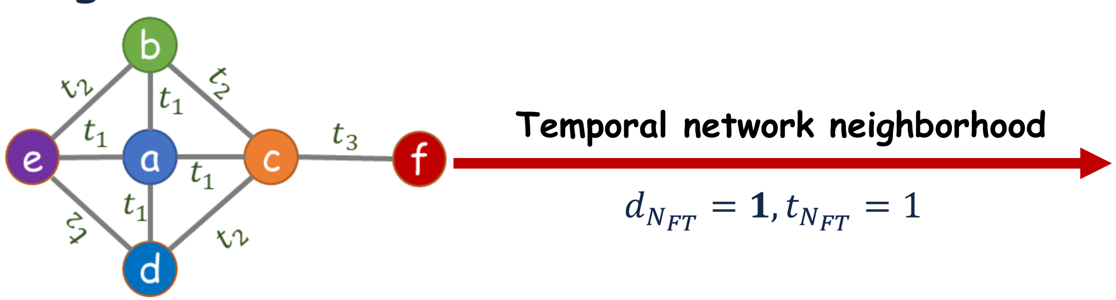
Temporal Network Neighborhood

- Definition 3. Temporal Network Neighborhood

  Given a temporal occurrence v^{t_v}, the neighborhood of v^{t_v} is defined as

$$N_{FT}(v^{t_v}) := \{v_i^{t_{v_i}} | f_{sp}\left(v_i^{t_{v_i}}, v^{t_v}\right) \leq d_{N_{FT}}, |t_v - t_{v_i}| \leq t_{N_{FT}}\}$$

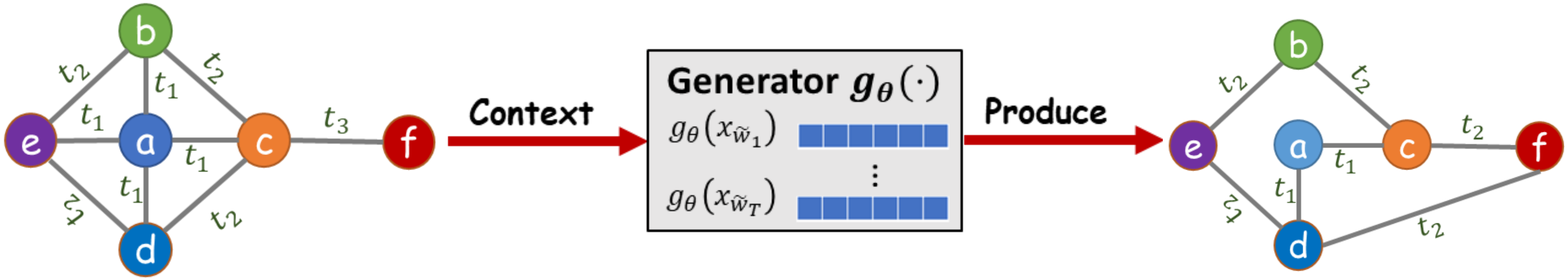
  where f_{sp} denotes the shortest path distance, $d_{N_{FT}}$ is the user-defined neighborhood range, and $t_{N_{FT}}$ refers to the user-defined neighborhood time window.



$\boldsymbol{d_{N_{FT}} = 1, t_{N_{FT}} = 1}$
• Representative node: v_a $\mathcal{N}_{FT}(v_a) = \{v_b, v_c, v_d, v_e\}$ • Underrepresented node: v_f $\mathcal{N}(v_f) = \{v_c\}$

Temporal Interaction Network Generation

- Problem 1. Temporal Interaction Network Generation
  - **Given:** a temporal interaction network $\tilde{G}$, which is represented as a collection of timestamped edges $\tilde{E} = (e_1^{t_{e_1}}, e_2^{t_{e_2}}, \dots, e_m^{t_{e_m}})$.
  - **Find:** a synthetic temporal interaction network $\tilde{G}'$ that accurately captures the structural and temporal properties of the observed temporal network $\tilde{G}$.



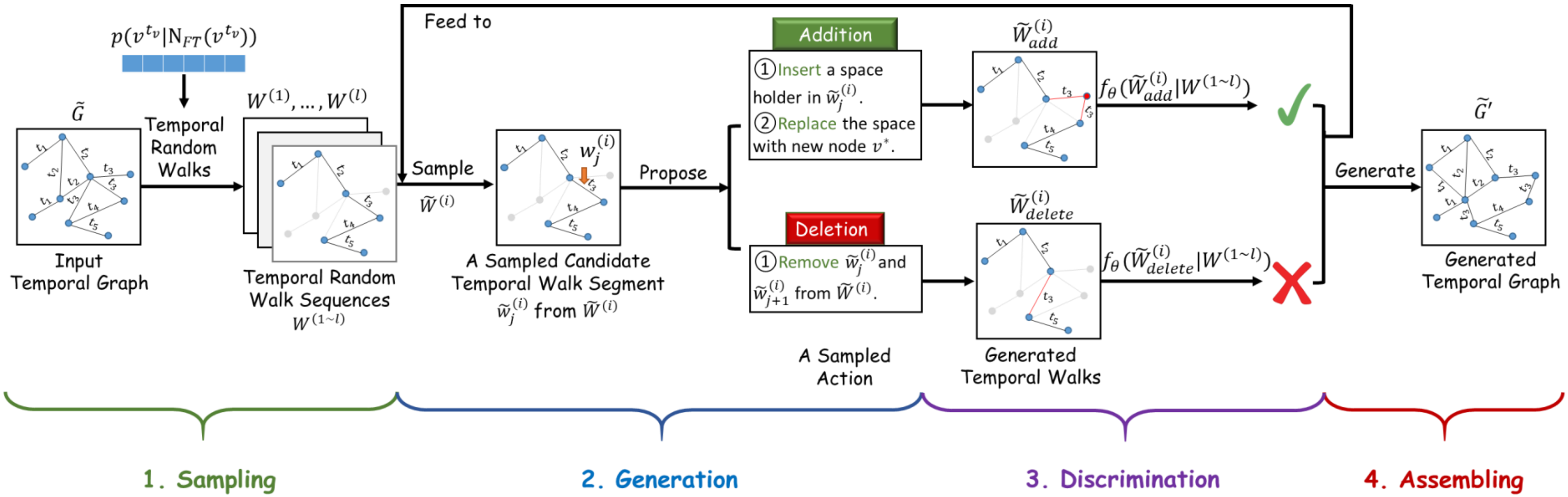
Original Graph $\tilde{G}$

Generated Graph $\tilde{G}'$

Roadmap

- Motivation
- Problem Definition
- Proposed TagGen Framework
- Experiments
- Conclusion

An Overview of TagGen



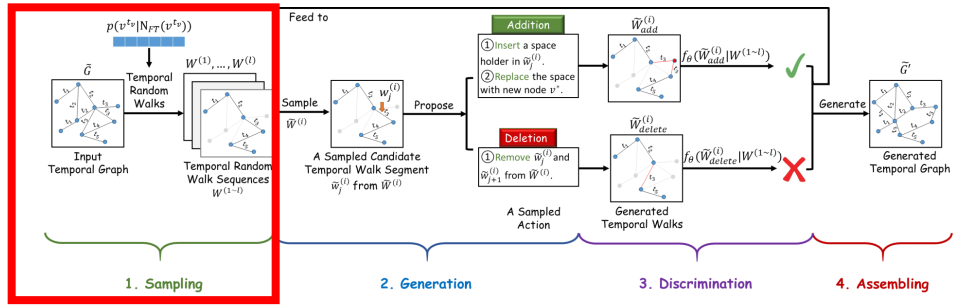
Our Innovations

1. A novel context extraction strategy for temporal interaction networks.

2. A family of local operations to perform addition and deletion of nodes and edges.

3. A bi-level self-attention mechanism.

TagGen Framework



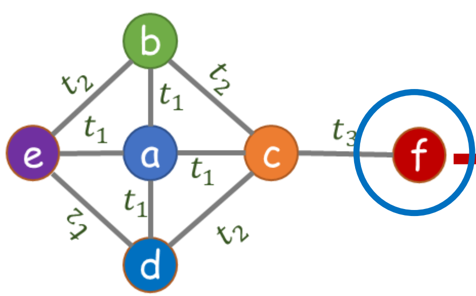
- S1: Context Sampling
  - Goal: select initial nodes for conducting temporal random walks.
  - Assumption: weak dependence.

Sampling distribution

Importance of v^{t_v}

For any $v^{t_v} \in \tilde{G}$, the corresponding temporal neighborhood distribution and topology neighborhood distribution satisfy a weak dependence, just in case, for δ>0,

$$p(v^{t_v}|\mathcal{N}_{FT}(v^{t_v})) \geq \delta[p(v^{t_v}|\mathcal{N}_{T}(v^{t_v}))p(v^{t_v}|\mathcal{N}_{S}(v^{t_v}))].$$



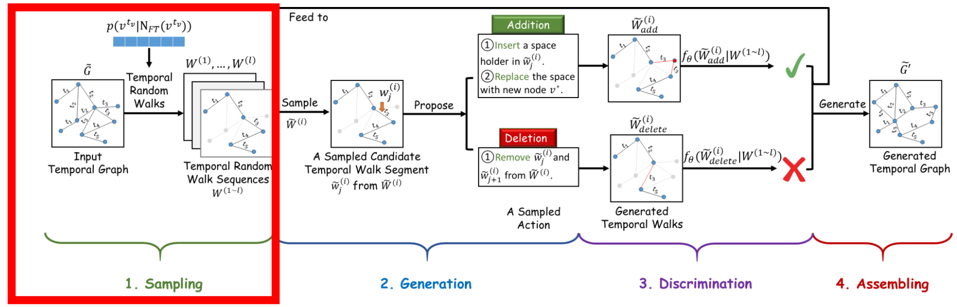
Temporal network neighborhood

$d_{N_{FT}} = 1, t_{N_{FT}} = 1$
• Representative node: v_a $\mathcal{N}_{FT}(v_a) = \{v_b, v_c, v_d, v_e\}$
• Underrepresented node: v_f $\mathcal{N}(v_f) = \{v_c\}$

$$p\left(v_f^{t_3} \middle| N_{FT}\left(v_f^{t_3}\right)\right) = p\left(v_f^{t_3} \middle| v_c^{t_2}\right)$$

Importance of $v_f^{t_3}$

TagGen Framework



- S1: Context Sampling
  - Solution: context sampling rule.

    Lemma 1. For any $v^{t_v} \in \tilde{G}$, if the corresponding temporal neighborhood distribution and topology neighborhood distribution satisfy a weak dependence, then the following inequality holds:

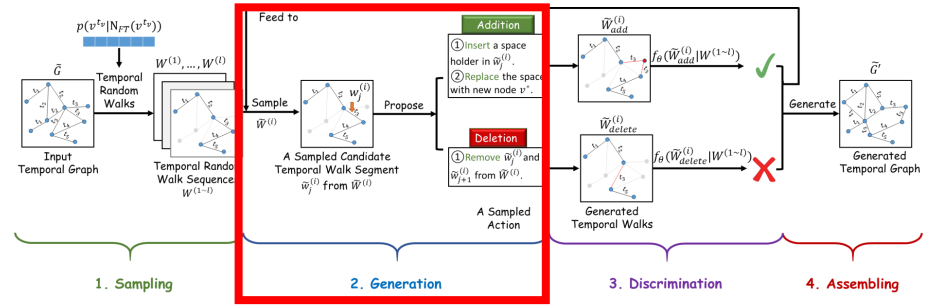
Sampling distribution

Uniform distribution

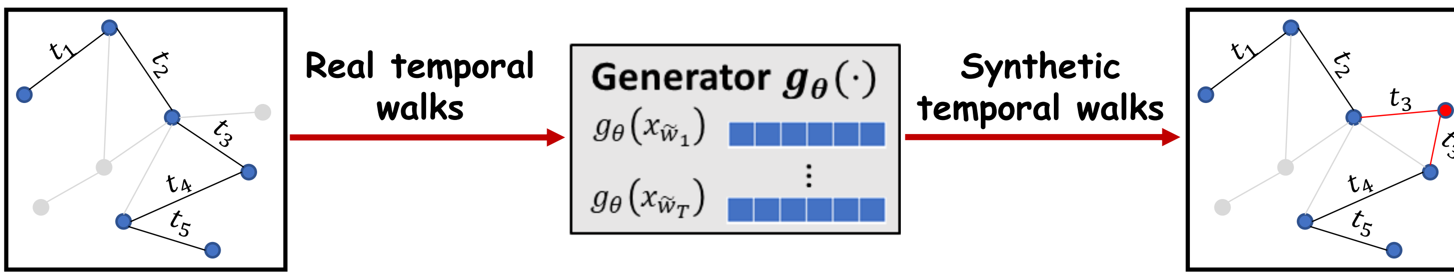
$$p(v^{t_v}|N_{FT}(v^{t_v})) \geq \alpha \frac{p(v^{t_v}|N_S(v^{t_v}))p(v^{t_v}|N_T(v^{t_v}))p(N_S(v^{t_v}))p(N_T(v^{t_v}))}{p(N_S(v^{t_v}), N_T(v^{t_v}))}$$

Kernel density estimation

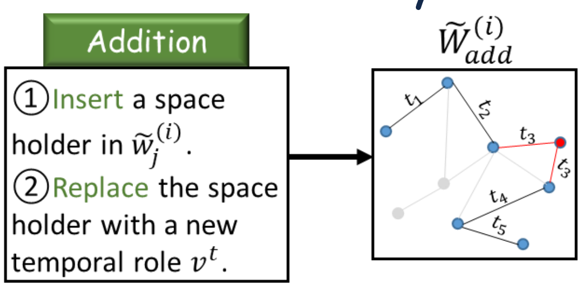
TagGen Framework



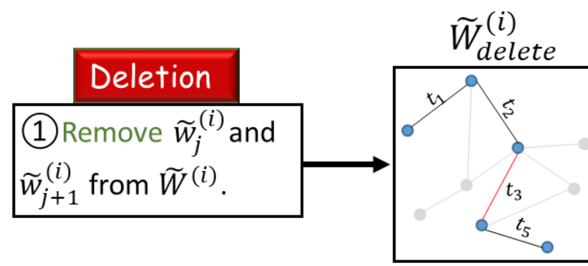
- S2: Sequence Generation
  - Goal: generate synthetic temporal random walks.



  - Solution: mimic dynamic network evolution via local operations.

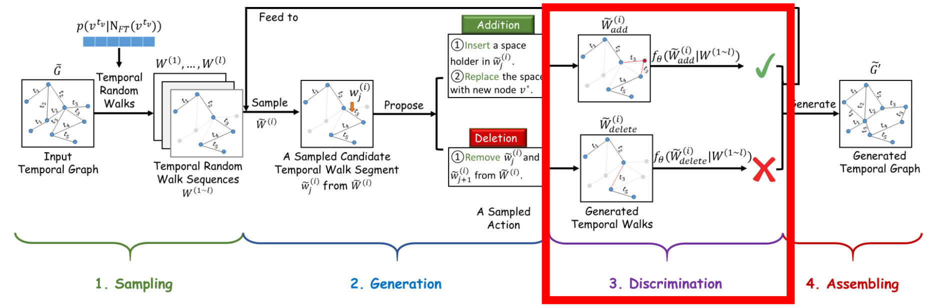


Add a temporal node



Remove a temporal node

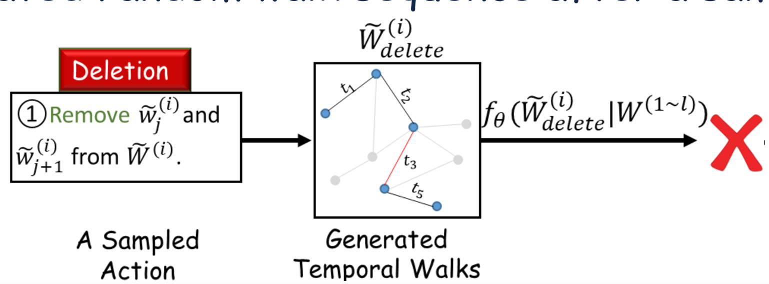
TagGen Framework



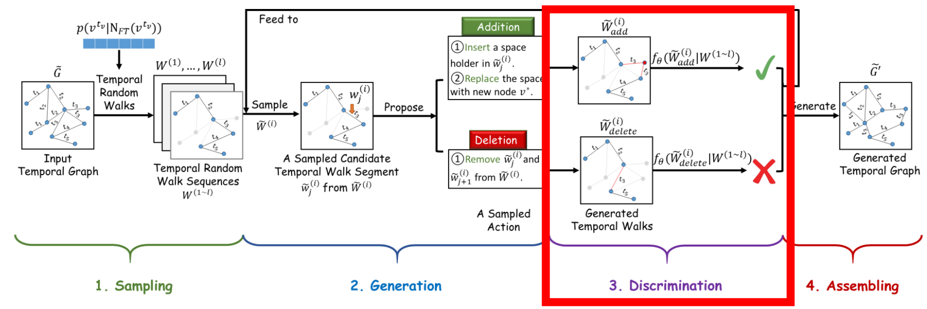
- S3: Sequence Discrimination
  - Goal: select synthetic random walks that are plausible in the input graph.
  - Solution: a bi-level self-attention mechanism.
    - maximize the action likelihood $p(\widetilde{W}^{(i)}{}_{action}|W^{(1\sim l)})$ via the deep autoregressive model $f_\theta(\cdot)$.

$$p(\widetilde{W}^{(i)}_{action}|W^{(1\sim l)}) \propto p_{action}(action) f_\theta(\widetilde{W}^{(i)}_{action})$$

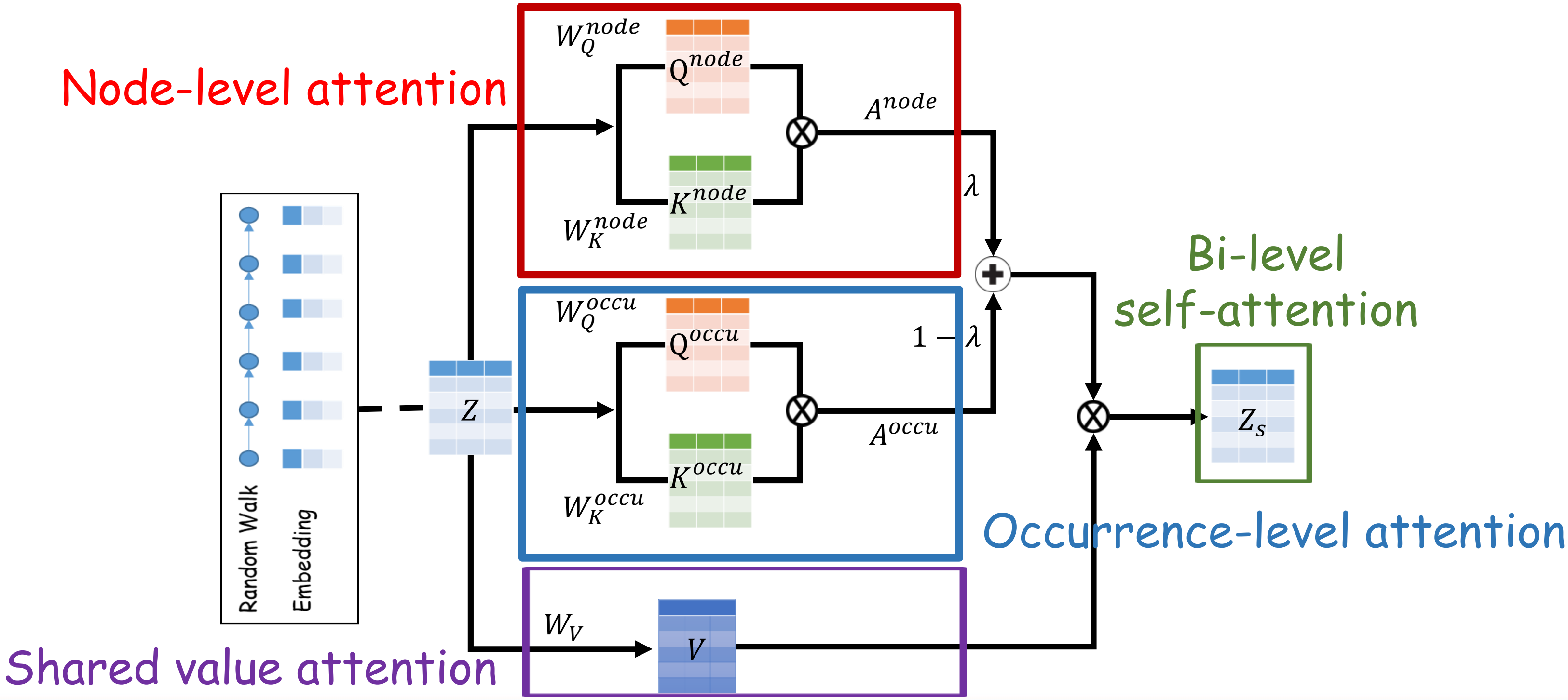
    - $p_{action} = \{p_{add}, p_{delete}\}$, where $p_{add} + p_{delete} = 1$.
    - $\widetilde{W}_{action}$: generated random walk sequence after a sampled action.



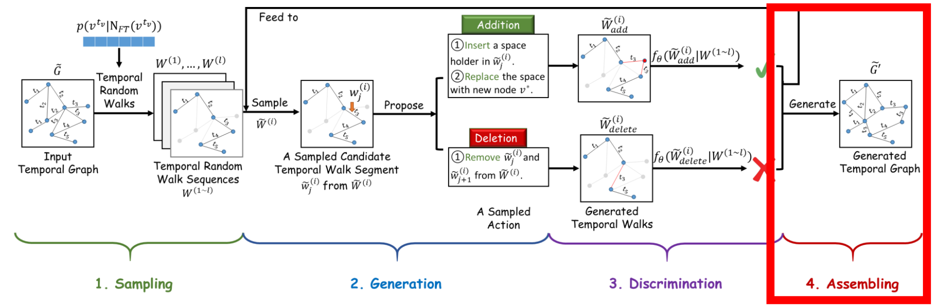
TagGen Framework



- S3: Sequence Discrimination
  - Solution: a bi-level self-attention mechanism.



TagGen Framework



- S4: Graph Assembling
  - Goal: assemble all the generated temporal random walks and generate the temporal interaction networks.
  - Solution: assembling rules to avoid some rare temporal occurrences (i.e., with a small degree) are not sampled.
    - Sample at least one temporal edge starting from each temporal node with probability $p(v^{t_v})$.
    - Sample at least one temporal edge at each timestamp with probability $p(e^{t_e})$.
    - Stop until the generated graph has the same edge density as the input graphs.

Roadmap

- Motivation
- Problem Definition
- Proposed *TagGen* Framework
- Experiments
- Conclusion

Experimental Setup

- Comparison Methods
  - Random graph models: E-R, B-A.
  - Temporal network embedding models: HTNE, DAE.
  - Deep graph generative models: GAE, NetGAN.
- Datasets

Network	Nodes	Temporal Edges	Timestamps
EMAIL	986	332,334	26
DBLP	1,909	8,237	15
WIKI	7,118	95,333	6
MSG	1,899	20,296	28
BITCOIN	3,783	24,186	117
SO	3,262	13,077	36
MO	13,840	195,330	20

Experimental Setup

- Network Properties for Evaluating Graph Generation

Metric name	Computation	Description
Mean Degree	$\mathbb{E}[d(v)]$	Mean degree of nodes in the graph.
Claw Count	$\sum_{v \in V} \binom{d(v)}{3}$	Number of the claw of the graph.
Wedge Count	$\sum_{v \in V} \binom{d(v)}{2}$	Number of wedges of the graph.
LCC	$\max_{f \in F} \|f\|$	Size of the largest connected component of the graph, where F is the set of all connected components in the graph.
PLE	$1 + n(\sum_{u \in V} \log(\frac{d(u)}{d_{min}}))^{-1}$	Exponent of the power-law distribution of the graph.
N-Component	$\lvert F \rvert$	Number of connected components, where F is the set of all connected components in the graph.

Experimental Setup

- Evaluation Metrics for Graph Generation
  - Original graph $\tilde{G} = \{\tilde{G}^{(1)}, \tilde{G}^{(2)}, \dots, \tilde{G}^{(T)}\}$.
  - Generated graph $\tilde{G}' = \{\widetilde{G'}^{(1)}, \widetilde{G'}^{(2)}, \dots, \widetilde{G'}^{(T)}\}$.
  - Selected network property $f_m(\cdot)$, e.g., Mean Degree, LCC.
  - Average discrepancy.

$$f_{avg}(\widetilde{G}, \widetilde{G'}, f_m) = Mean_{t=1:T}\left(\left|\frac{f_m(\widetilde{S}^t) - f_m(\widetilde{S'}^t)}{f_m(\widetilde{S}^t)}\right|\right)$$

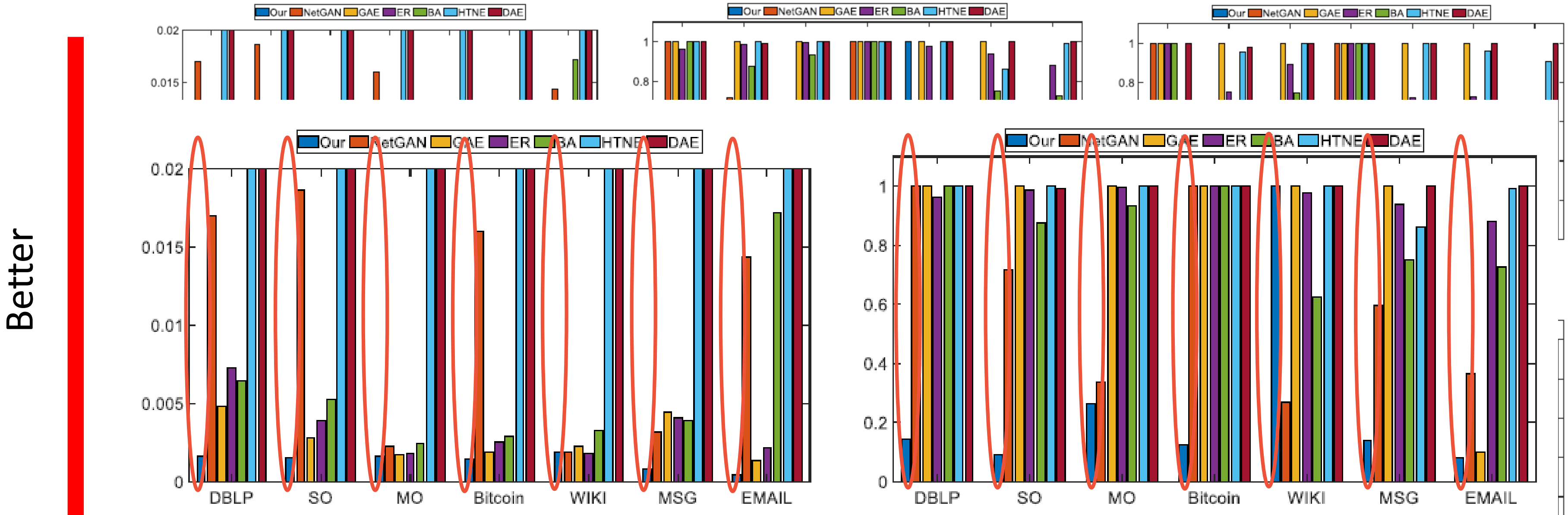
  - Median discrepancy.

$$f_{med}(\widetilde{G}, \widetilde{G'}, f_m) = Median_{t=1:T}\left(\left|\frac{f_m(\widetilde{S}^t) - f_m(\widetilde{S'}^t)}{f_m(\widetilde{S}^t)}\right|\right)$$

Temporal Interaction Network Generation

$$f_{avg}(\widetilde{G}, \widetilde{G'}, f_m) = Mean_{t=1:T}\left(\left|\frac{f_m(\widetilde{S}^t) - f_m(\widetilde{S'}^t)}{f_m(\widetilde{S}^t)}\right|\right)$$

- Quantitative Evaluation in Average Discrepancy



(a) Mean Degree

(b) Claw Count

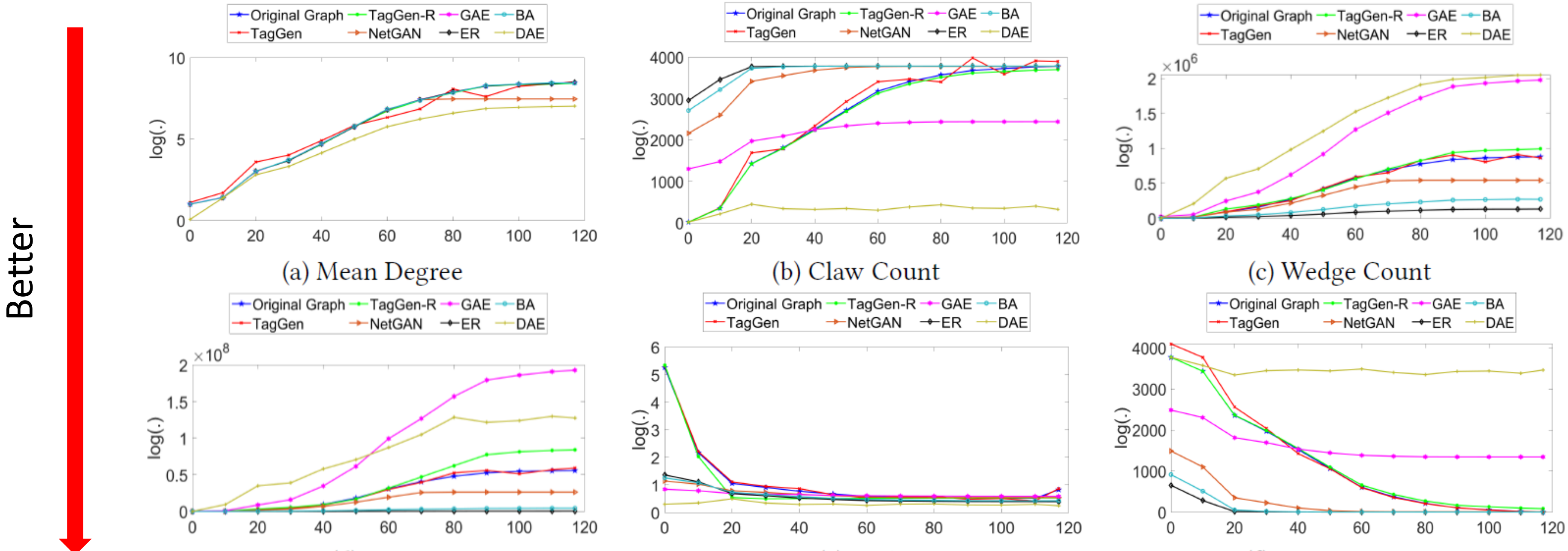
(d) LCC

(e) PLE

(f) N-Component

Temporal Interaction Network Generation

- Fine-Grained Quantitative Evaluation in BITCOIN across 117 Timestamps



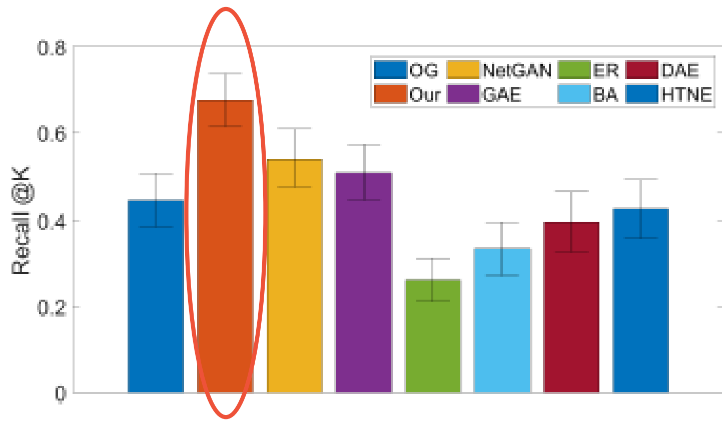
(a) Mean Degree (b) Claw Count (c) Wedge Count

(d) LCC (e) PLE (f) N-Component

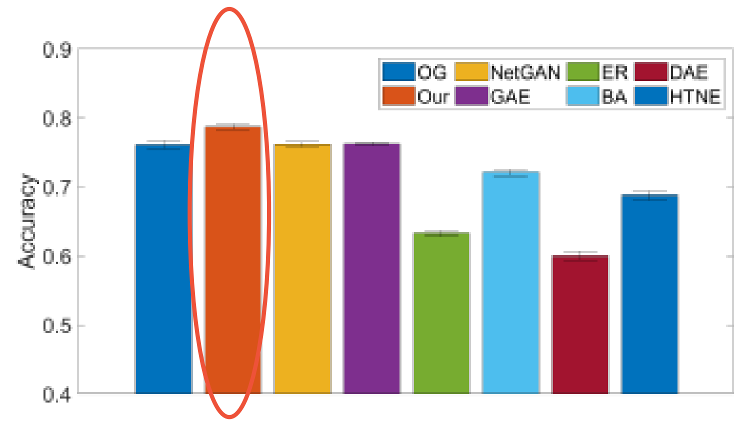
Data Augmentation

- Data Augmentation in the Task of Anomaly Detection and Link Prediction

Better



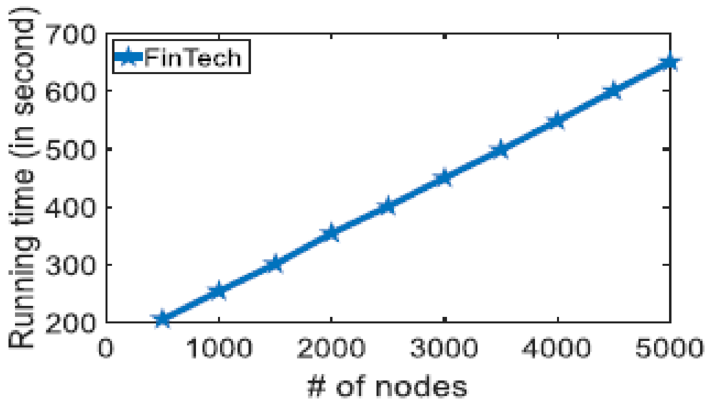
(a) Anomaly detection



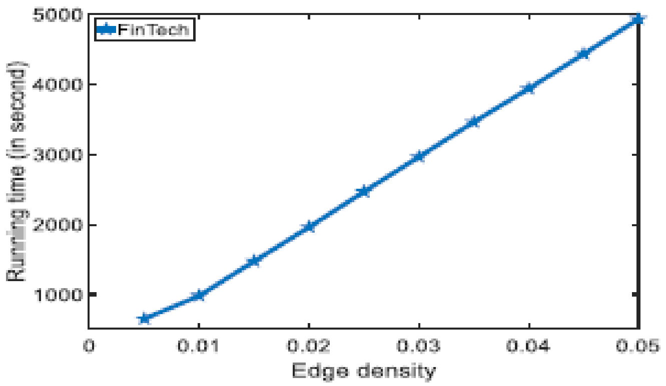
(b) Link prediction

Scalability Analysis

- Scalability Analysis w.r.t. Controlled Increasing # of Nodes and Edge Density



(a)Running Time v.s # of nodes

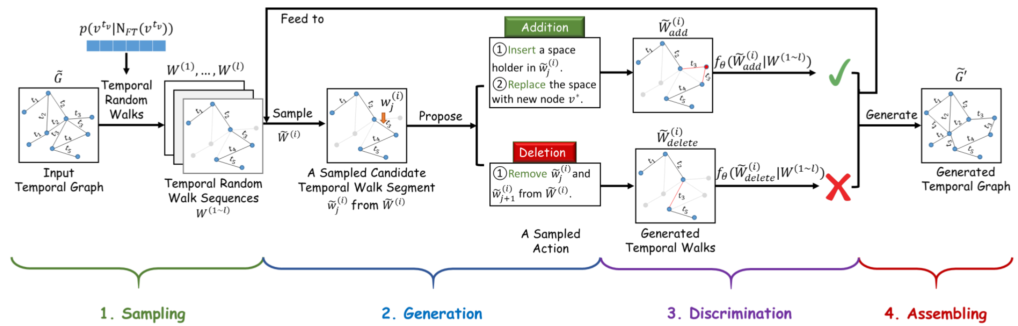


(b) Running Time v.s edge density

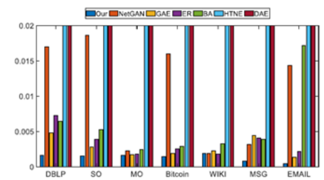
Roadmap

- Motivation
- Problem Definition
- Proposed *TagGen* Framework
- Experiments
- Conclusion

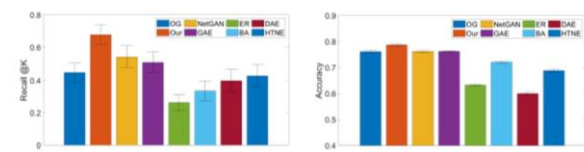
Conclusion



- Technical Innovations
  - A novel context extraction strategy for temporal interaction networks.
  - A bi-level self-attention mechanism to ensure quality of the generated temporal graph.

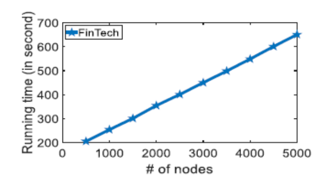


(a) Mean Degree



(a) Anomaly detection (b) Link prediction

- Results
  - TagGen outperforms baseline methods in the tasks of temporal interaction network generation and data augmentation.
  - TagGen runs in linear time w.r.t. the size of graphs.



(a)Running Time v.s # of nodes

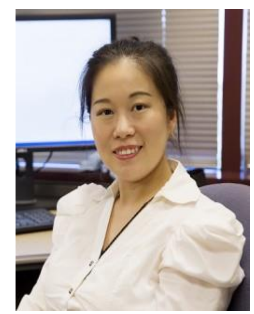
AUGUST 23-27th
KDD2020
Virtual Conference

Thank You!

Dawei Zhou
(UIUC)

Lecheng Zheng
(UIUC)

Jiawei Han
(UIUC)



Jingrui He
(UIUC)

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Paper: https://sites.google.com/view/dawei-zhou/publications?authuser=0

Data and code: https://github.com/davidchouzdw/TagGen