KDD-24 Research Track Paper

Self-Explainable Temporal Graph Networks based on Graph Information Bottleneck

Sangwoo Seo, Sungwon Kim, Jihyeong Jung,
Yoonho Lee, Chanyoung Park

Korea Advanced Institute of Science and Technology (KAIST)

DSAIL @KAIST

EXPLANATION FOR TEMPORAL GRAPH

- **Temporal Graph Models**
  - Temporal graph models can predict the occurrence of target events based on past events.

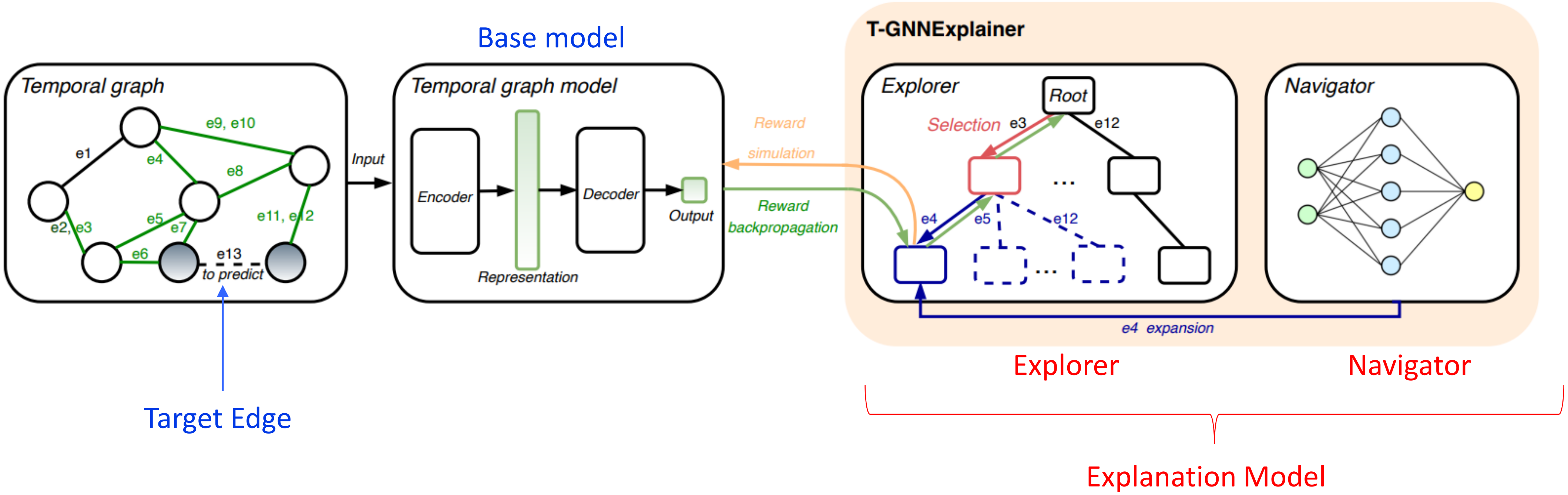
- **Need for Explanation Models in Temporal Graphs**
  - Temporal graph models are often considered as black boxes because they cannot identify how past events influence outcomes.
  - Increased reliability and transparency in predictions.

- **Objective of Temporal Graph Explanations**
  - The goal of an explanation model for temporal graphs is to detect past events that are important for predicting the occurrence of the target event.

T-GNNEXPLAINER

Explaining Temporal Graph Models through an Explorer-Navigator Framework

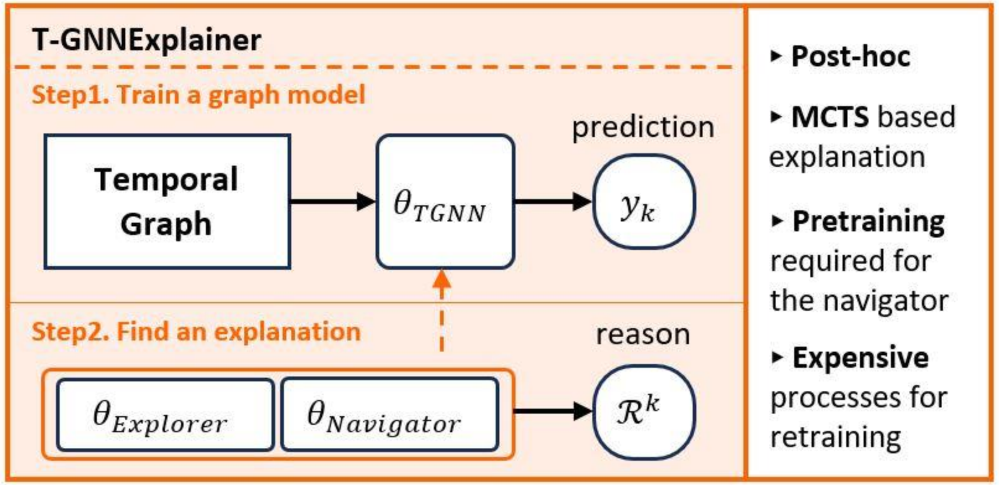
- Post-hoc explanation model
  - Explanations are generated based on a pretrained base model.
- Require separate models for prediction and explanation.



Xia, Wenwen, et al. "Explaining temporal graph models through an explorer-navigator framework." The Eleventh International Conference on Learning Representations. 2023.

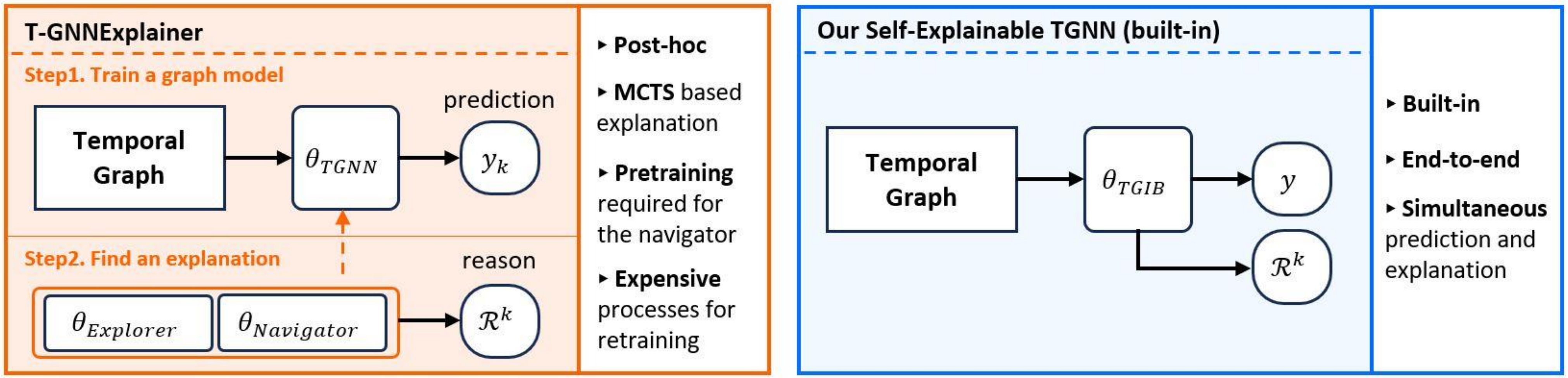
MOTIVATION

- Major Drawback of **Post-hoc models**
  1) Explanation model needs **frequent retraining** based on the retrained base model.
  2) Examining the behavior of an already trained base model can be **challenging to fully comprehend the base model.**



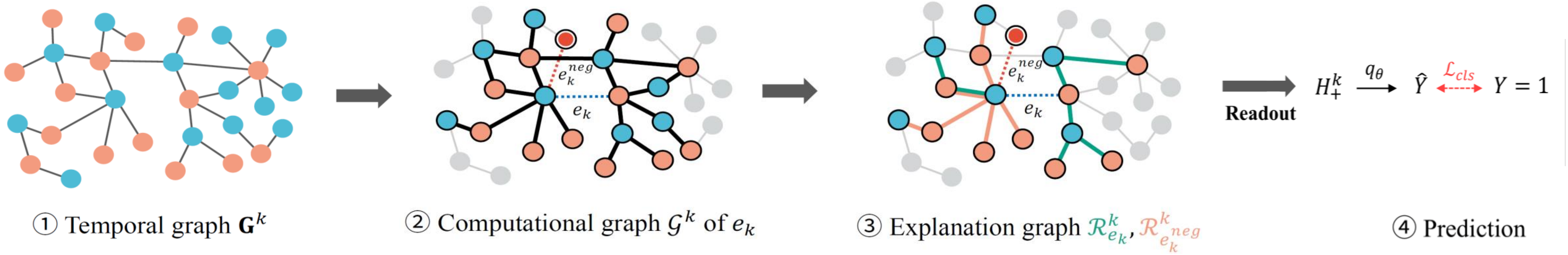
MOTIVATION

- Our Self-Explainable Model
  - **Built-in explanation framework** for temporal graphs.
    - End-to-end model for temporal graphs that **generates predictions and explanations simultaneously**.



OVERALL PROCESS

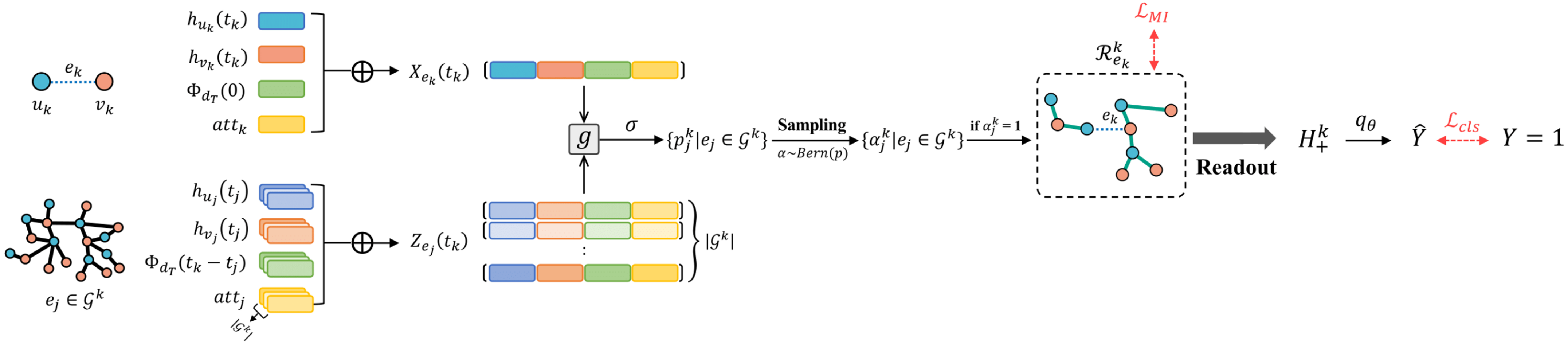
- Our Self-Explainable Model
  1. Define a temporal graph as the collection of past events that occurred before the current time.
  2. Use L-hop computational subgraph of target event as candidate graph.
  3. Extract important candidate events as explanation graph.
  4. Predict the occurrence of the target event based on explanation graph.



METHODOLOGY

- **Main idea**
  - Considers the **interaction between the target event and candidate events** to extract important candidate events.
  - Utilize the **Information Bottleneck (IB)** approach
    - **Control information flow** from candidate events to predictions by introducing stochasticity into edges.

    → Focus on the most relevant information for making accurate predictions



METHODOLOGY

- **GIB-based Objective for Temporal Graph**
  - We extract explanation graph $\mathcal{R}^k$ for the target edge e_k from its L-hop neighborhood $\mathcal{G}^k$.
  - $\mathcal{R}^k$ is a subgraph of e_k's L-hop computation graph $\mathcal{G}^k$.
  - Y_k is the label information indicating the occurrence of the event.

$$\min_{\mathcal{R}^k} \underline{-I\left(Y_k; \mathcal{R}^k\right)} + \beta \underline{I\left(\mathcal{R}^k; e_k, \mathcal{G}^k\right)}$$

1. Sufficiently learn label-relevant information
2. $\mathcal{R}^{\mathrm{k}}$ efficiently includes only important information related to e_k and $\mathcal{G}^k$

METHODOLOGY

- **GIB-based Objective for Temporal Graph**
  - We obtain an upper bound on each term to optimize the objective function.

$$\min_{\mathcal{R}^k} \underline{-I\left(Y_k;\mathcal{R}^k\right)} + \underline{\beta\, I\left(\mathcal{R}^k; e_k, \mathcal{G}^k\right)}$$

$$\begin{aligned}
-I(Y_k;\mathcal{R}^k) &= \mathbb{E}_{Y_k,e_k,\mathcal{R}^k}\left[-\log\frac{p(Y_k,\mathcal{R}^k)}{p(Y_k)p(\mathcal{R}^k)}\right] \\
&= \mathbb{E}_{Y_k,e_k,\mathcal{R}^k}\left[-\log\frac{p(Y_k|\mathcal{R}^k)}{p(Y_k)}\right] \\
&= \mathbb{E}_{Y_k,e_k,\mathcal{R}^k}\left[-\log p\left(Y_k|\mathcal{R}^k\right)\right] + \mathbb{E}_{Y_k}\left[\log p(Y_k)\right] \\
&= \mathbb{E}_{Y_k,e_k,\mathcal{R}^k}\left[-\log p\left(Y_k|\mathcal{R}^k\right)\right] - H(Y) \\
&\le \mathbb{E}_{Y_k,e_k,\mathcal{R}^k}\left[-\log q_\theta\left(Y_k|\mathcal{R}^k\right)\right] - H(Y) \\
&\le \mathbb{E}_{Y_k,e_k,\mathcal{R}^k}\left[-\log q_\theta\left(Y_k|\mathcal{R}^k\right)\right] = \mathcal{L}_{cls}
\end{aligned}$$

$$\begin{aligned}
I(\mathcal{R}^k; e_k, \mathcal{G}^k) &= \mathbb{E}_{\mathcal{R}^k,e_k,\mathcal{G}^k}\left[\log\frac{p(\mathcal{R}^k; e_k, \mathcal{G}^k)}{p(\mathcal{R}^k)p(e_k,\mathcal{G}^k)}\right] \\
&= \mathbb{E}_{\mathcal{R}^k,e_k,\mathcal{G}^k}\left[\log\frac{p(\mathcal{R}^k|e_k,\mathcal{G}^k)}{p(\mathcal{R}^k)}\right] \\
&= \mathbb{E}_{\mathcal{R}^k,e_k,\mathcal{G}^k}\left[\log p\left(\mathcal{R}^k|e_k,\mathcal{G}^k\right) - \log p\left(\mathcal{R}^k\right)\right] \\
&\le \mathbb{E}_{\mathcal{R}^k,e_k,\mathcal{G}^k}\left[\log p\left(\mathcal{R}^k|e_k,\mathcal{G}^k\right) - \log q\left(\mathcal{R}^k\right)\right] \\
&\le \mathbb{E}_{e_k,\mathcal{G}^k}\left[\mathrm{KL}\left[p\left(\mathcal{R}^k|e_k,\mathcal{G}^k\right) \parallel q\left(\mathcal{R}^k\right)\right]\right] = \mathcal{L}_{MI}
\end{aligned}$$

METHODOLOGY

- **Time-aware event representation**
  - For the self-attention mechanism, we define the query, key and value as:

Query   Node embedding   Edge attribute   Time encoding

$$Q^{(l)}(t) = \left[h_z^{(l-1)}(t) \| att_{z,0} \| \Phi_{d_T}(0) \right]$$

Key

$$K^{(l)}(t) = \begin{bmatrix} K_1^{(l)}(t) \\ \vdots \\ K_n^{(l)}(t) \end{bmatrix} = \begin{bmatrix} h_{z_1}^{(l-1)}(t_{z,1}) \| att_{z,1} \| \Phi_{d_T}(t - t_{z,1}) \\ \vdots \\ h_{z_n}^{(l-1)}(t_{z,n}) \| att_{z,n} \| \Phi_{d_T}(t - t_{z,n}) \end{bmatrix}$$

Value

$$V^{(l)}(t) = \begin{bmatrix} V_1^{(l)}(t) \\ \vdots \\ V_n^{(l)}(t) \end{bmatrix} = \begin{bmatrix} h_{z_1}^{(l-1)}(t_{z,1}) \| att_{z,1} \| \Phi_{d_T}(t - t_{z,1}) \\ \vdots \\ h_{z_n}^{(l-1)}(t_{z,n}) \| att_{z,n} \| \Phi_{d_T}(t - t_{z,n}) \end{bmatrix}$$

  - Each node collects information from its neighboring nodes, and the attention weights are defined as:

Attention weight

$$\alpha_i^{(l)}(t) = \mathrm{softmax}\left(\frac{Q^{(l)}(t) K_i^{(l)}(t)^T}{\sqrt{(d + f_{\text{edge}} + d_T)}} \right)$$

  - Finally, we obtain the hidden neighborhood representations as follows:

Neighborhood representation

$$\tilde{h}_z^{(l)}(t) = \mathrm{Attn}\left(Q^{(l)}(t), K^{(l)}(t), V^{(l)}(t) \right) = \mathrm{softmax}\left(\frac{Q^{(l)}(t) K^{(l)}(t)^T}{\sqrt{(d + f_{\text{edge}} + d_T)}} \right) V^{(l)}(t)$$

METHODOLOGY

- **Time-aware event representation**
  - We concatenate the neighborhood representation with the node feature, and use it as the input to a feed-forward network as follows:

$$\underbrace{h_z^{(l)}(t)}_{\text{Final node embedding}} = \mathrm{FFN}\left(\left[\underbrace{\tilde{h}_z^{(l)}(t)}_{\text{Neighborhood representation}} \| \underbrace{x_z}_{\text{Node feature}}\right]\right)$$

$$= \mathrm{ReLU}\left(\left[\tilde{h}_z^{(l)}(t) \| x_z\right] W_0^{(l)} + b_0^{(l)}\right) W_1^{(l)} + b_1^{(l)}$$

  - We construct the time-aware event representation for the target event and candidate event.

Target Event $$X_{e_k}(t_k) = [\, \underbrace{h_{u_k}(t_k) \,\|\, h_{v_k}(t_k)}_{\text{Node representations}} \,\|\, \underbrace{\Phi_{d_T}(0)}_{\text{Time encoding}} \,\|\, \underbrace{\mathrm{att}_k}_{\text{Edge attribute}} \,]$$

Candidate Event $$Z_{e_j}(t_k) = [\, h_{u_j}(t_j) \,\|\, h_{v_j}(t_j) \,\|\, \Phi_{d_T}(t_k - t_j) \,\|\, \mathrm{att}_j]$$

METHODOLOGY

- **Minimizing $I(\mathcal{R}^k; e_k, \mathcal{G}^k)$**
  - We obtain upper bound of $I(\mathcal{R}^k; e_k, \mathcal{G}^k)$.

$$I(\mathcal{R}^k; e_k, \mathcal{G}^k) \leq \mathbb{E}_{e_k,\mathcal{G}^k}\left[\mathrm{KL}\left[p\left(\mathcal{R}^k | e_k, \mathcal{G}^k\right) \| q\left(\mathcal{R}^k\right)\right]\right]$$

  - We decompose $p(\mathcal{R}^k | e_k, \mathcal{G}^k)$ into a multivariate Bernoulli distribution.

$$p(\mathcal{R}^k | e_k, \mathcal{G}^k) = \prod_{e_j \in \mathcal{R}_k} \overbrace{p_j^k}^{p(e_j | e_k, \mathcal{G}^k)} \cdot \prod_{e_j \in \mathcal{G}^k \setminus \mathcal{R}_k} (1 - p_j^k)$$

  - Each p_j^k is computed as the output of an MLP that takes X_{e_k} and Z_{e_j} as input.

Importance score   Target event   Candidate event

$$p_j^k = p(e_j | e_k, \mathcal{G}^k) = \sigma\left(g\left(X_{e_k}(t_k),\ Z_{e_j}(t_k)\right)\right)$$

METHODOLOGY

- **Minimizing $I(\mathcal{R}^k; e_k, \mathcal{G}^k)$**
  - We use a multivariate Bernoulli distribution for $q(\mathcal{R}^k)$.

$$q(\mathcal{R}^k) = r^{|\mathcal{R}^k|}(1-r)^{|\mathcal{G}^k|-|\mathcal{R}^k|}$$

  - Finally, we specify upper bound of $I(\mathcal{R}^k; e_k, \mathcal{G}^k)$ and define the mutual information loss $\mathcal{L}_{\mathrm{MI}}$.

$$I(\mathcal{R}^k; e_k, \mathcal{G}^k) \le \mathbb{E}_{e_k,\mathcal{G}^k}\left[\mathrm{KL}\left[p\left(\mathcal{R}^k|e_k,\mathcal{G}^k\right)\|q\left(\mathcal{R}^k\right)\right]\right]$$
$$= \mathbb{E}_{p(e_k,\mathcal{G}^k)}\left[\sum_{e_j\in\mathcal{G}^k} p_j^k \log\frac{p_j^k}{r} + (1-p_j^k)\log\frac{1-p_j^k}{1-r}\right]$$

$$\begin{aligned}
\mathrm{I}(\mathcal{R}^k; e_k, \mathcal{G}^k) &\le \mathbb{E}_{e_k,\mathcal{G}^k}\left[\mathrm{KL}[p(\mathcal{R}^k|e_k,\mathcal{G}^k) \parallel q(\mathcal{R}^k)]\right] \\
&= \mathbb{E}_{e_k,\mathcal{G}^k}\left[\log\frac{p(\mathcal{R}^k|e_k,\mathcal{G}^k)}{q(\mathcal{R}^k)}\right] \\
&= \mathbb{E}_{e_k,\mathcal{G}^k}\left[\log\left(\frac{\prod_{e_j\in\mathcal{R}^k} p_j^k \cdot \prod_{e_j\in\mathcal{G}^k\setminus\mathcal{R}^k}(1-p_j^k)}{r^{|\mathcal{R}^k|}\cdot(1-r)^{|\mathcal{G}^k|-|\mathcal{R}^k|}}\right)\right] \\
&= \mathbb{E}_{e_k,\mathcal{G}^k}\left[\log\frac{\prod_{e_j\in\mathcal{R}^k} p_j^k}{r^{|\mathcal{R}^k|}} + \log\frac{\prod_{e_j\in\mathcal{G}^k\setminus\mathcal{R}^k}(1-p_j^k)}{(1-r)^{|\mathcal{G}^k|-|\mathcal{R}^k|}}\right] \\
&= \mathbb{E}_{e_k,\mathcal{G}^k}\left[\sum_{e_j\in\mathcal{R}^k}\log\frac{p_j^k}{r} + \sum_{e_j\in\mathcal{R}^k}\log\frac{1-p_j^k}{1-r}\right] \\
&= \mathbb{E}_{e_k,\mathcal{G}^k}\left[\sum_{e_j\in\mathcal{G}^k} p_j^k\log\frac{p_j^k}{r} + (1-p_j^k)\log\frac{1-p_j^k}{1-r}\right]
\end{aligned}$$

METHODOLOGY

- **Minimizing** $-I(Y_k;\mathcal{R}^k)$
  - We sample stochastic weights from the Bernoulli distribution and obtain a valid event representation.

$$\underbrace{\tilde{Z}_{e_j}(t_k)}_{\text{Valid event representation}} = \underbrace{\alpha_j^k}_{\text{Stochastic weight}} \underbrace{Z_{e_j}(t_k)}_{\text{Candidate event}}, \quad \alpha_j^k \sim Ber(p_j^k)$$

  - We obtain the representation of $\mathcal{R}^k$ extracted from each valid event representation.

$$\underbrace{H_+^k}_{\mathcal{R}^k \text{ representation}} = \text{Readout}\left[\left\{\tilde{Z}_{e_j}(t_k)|e_j \in \mathcal{G}^k\right\}\right]$$

  - **Negative sample**
    - We fix the node u and replace node v by randomly sampling a node from the entire graph.

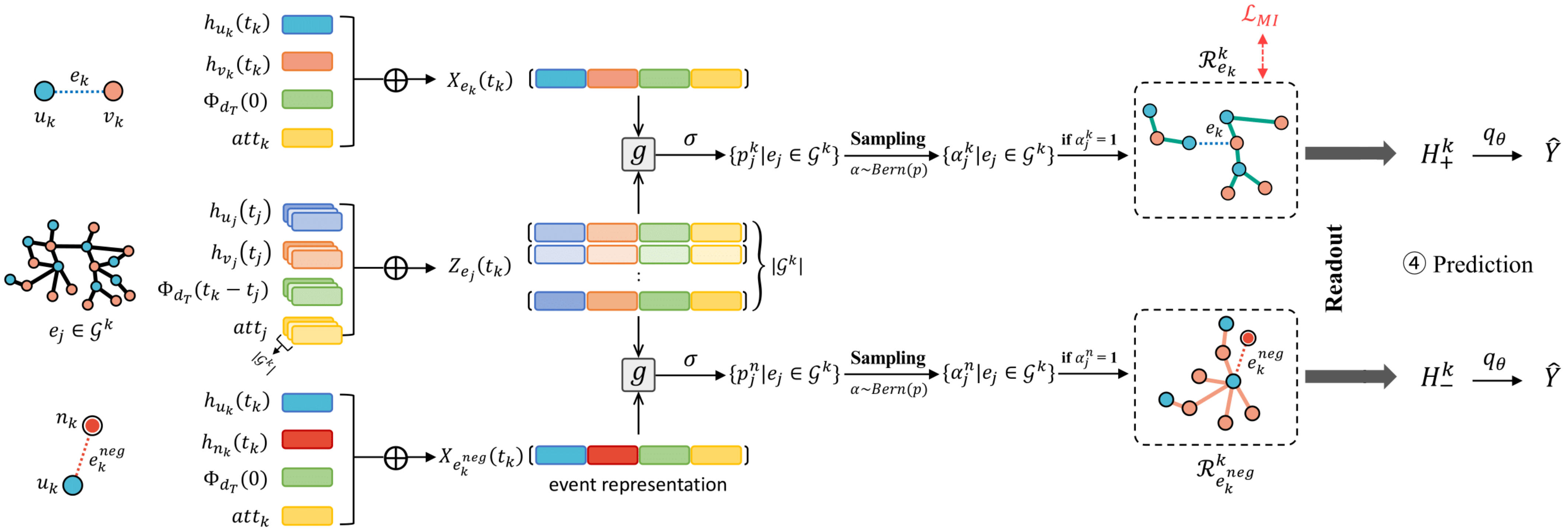
$$\underbrace{X_{e_k}^{\text{neg}}(t_k)}_{\text{Negative sample representation}} = \left[\ h_{u_k}(t_k) \parallel h_{n_k}(t_k) \parallel \Phi_{d_T}(0) \parallel att_k\ \right]$$

  - Finally, we use the time-aware link prediction loss function

$$\mathcal{L}_{\text{cls}} = \sum_{e_k \in S} -\log\left[\sigma\left(\underbrace{q_\theta}_{\text{MLP}}(\underbrace{X_{e_k}}_{\text{Target event}}, \underbrace{H_+^k}_{\mathcal{R}^k})\right)\right] - N \cdot \mathbb{E}_{\text{neg}\sim P_n} \log\left[\sigma\left(\underbrace{q_\theta}_{\text{MLP}}(\underbrace{X_{e_k}^{\text{neg}}}_{\text{Negative sample}}, \underbrace{H_-^k}_{\mathcal{R}_{e^{neg}}^k})\right)\right]$$

METHODOLOGY

- **Architecture**



EXPERIMENTS

- **Datasets**

Dataset	Domain	#Nodes	#Edges	#Edge Features	Duration
Wikipedia	Social	9,227	157,474	172	1 month
UCI	Social	1,899	58,835	-	196 days
USLegis	Politics	225	60,396	1	12 terms
CanParl	Politics	734	74,478	1	14 years
Enron	Social	184	125,235	-	3 years
Reddit	Social	10,984	672,447	172	1 month

- **Baselines**
  - **Link prediction**
    - Jodie
    - DyRep
    - TGAT
    - TGN
    - TGL
    - CAW-N
    - GraphMixer
  - **Explanation performance**
    - ATTN
    - Grad-CAM
    - GNNExplainer
    - PGExplainer
    - T-GNNExplainer

EXPERIMENTS

- **Link Prediction**

	Model	Wikipedia	UCI	USLegis	CanParl	Enron	Reddit
Transductive	Jodie	94.62 ± 0.50	86.73 ± 1.00	73.31 ± 0.40	69.26 ± 0.31	77.31 ± 4.20	97.11 ± 0.30
	DyRep	92.43 ± 0.37	53.67 ± 2.10	57.28 ± 0.71	54.02 ± 0.76	74.55 ± 3.95	96.09 ± 0.11
	TGAT	95.34 ± 0.10	73.01 ± 0.60	68.89 ± 1.30	70.73 ± 0.72	68.02 ± 0.10	98.12 ± 0.20
	TGN	97.58 ± 0.20	80.40 ± 1.40	<u>75.13 ± 1.30</u>	70.88 ± 2.34	79.91 ± 1.30	<u>98.30 ± 0.20</u>
	TCL	96.47 ± 0.16	89.57 ± 1.63	69.59 ± 0.48	68.67 ± 2.67	79.70 ± 0.71	97.53 ± 0.02
	CAW-N	<u>98.28 ± 0.20</u>	90.03 ± 0.40	69.94 ± 0.40	69.82 ± 2.34	**89.56 ± 0.09**	97.95 ± 0.20
	GraphMixer	97.25 ± 0.03	<u>93.25 ± 0.57</u>	70.74 ± 1.02	<u>77.04 ± 0.46</u>	82.25 ± 0.16	97.31 ± 0.01
	TGIB	**99.37 ± 0.09**	**93.60 ± 0.24**	**91.61 ± 0.34**	**87.07 ± 0.44**	<u>82.42 ± 0.11</u>	**99.68 ± 0.15**

	Model	Wikipedia	UCI	USLegis	CanParl	Enron	Reddit
Inductive	Jodie	93.11 ± 0.40	71.23 ± 0.80	52.16 ± 0.50	53.92 ± 0.94	76.48 ± 3.50	94.36 ± 1.10
	DyRep	92.05 ± 0.30	50.43 ± 1.20	56.26 ± 2.00	54.02 ± 0.76	66.97 ± 3.80	95.68 ± 0.20
	TGAT	93.82 ± 0.30	66.89 ± 0.40	52.31 ± 1.50	55.18 ± 0.79	63.70 ± 0.20	96.42 ± 0.30
	TGN	97.05 ± 0.20	74.70 ± 0.90	<u>58.63 ± 0.37</u>	54.10 ± 0.93	77.94 ± 1.02	96.87 ± 0.20
	TCL	96.22 ± 0.17	87.36 ± 2.03	52.59 ± 0.97	54.30 ± 0.66	76.14 ± 0.79	94.09 ± 0.07
	CAW-N	<u>97.70 ± 0.20</u>	89.65 ± 0.40	53.11 ± 0.40	55.80 ± 0.69	**86.35 ± 0.51**	<u>97.37 ± 0.30</u>
	GraphMixer	96.65 ± 0.02	<u>91.19 ± 0.42</u>	50.71 ± 0.76	<u>55.91 ± 0.82</u>	75.88 ± 0.48	95.26 ± 0.02
	TGIB	**99.28 ± 0.11**	**91.26 ± 0.16**	**86.42 ± 0.16**	**79.56 ± 0.79**	<u>80.64 ± 0.59</u>	**99.54 ± 0.02**

Setup

- Inductive Setting
  - We predict the occurrence of events including nodes not observed during the training time.
- Transductive Setting
  - We predict the occurrence of events including both observed and unobserved nodes during training time.

Observation

- TGIB demonstrated the significant performance compared to the baselines for the temporal graphs in both transductive and inductive settings.

EXPERIMENTS

- **Explanation Performance**

	Wikipedia	UCI	USLegis	CanParl	Enron
Random	70.91 ± 1.03	54.51 ± 0.52	54.24 ± 1.34	51.66 ± 2.26	48.94 ± 1.28
ATTN	77.31 ± 0.01	27.25 ± 0.01	62.24 ± 0.00	79.92 ± 0.01	68.28 ± 0.01
Grad-CAM	83.11 ± 0.01	26.06 ± 0.01	78.98 ± 0.01	50.42 ± 0.01	19.93 ± 0.01
GNNExplainer	84.34 ± 0.16	62.38 ± 0.46	89.42 ± 0.50	80.59 ± 0.58	77.82 ± 0.88
PGExplainer	84.26 ± 0.78	59.47 ± 1.68	<u>91.42 ± 0.94</u>	75.92 ± 1.12	62.37 ± 3.82
T-GNNExplainer	<u>85.74 ± 0.56</u>	<u>68.26 ± 2.62</u>	90.37 ± 0.84	<u>80.67 ± 1.49</u>	<u>82.02 ± 1.94</u>
TGIB	**88.09 ± 0.68**	**87.06 ± 1.04**	**93.33 ± 0.72**	**89.72 ± 1.18**	**83.55 ± 0.91**

<u>Setup</u>

- To evaluate the performance of explanations, we measure the proportion of generated explanations that have the same predicted label as the original prediction.
- We evaluate the explanation performance over various sparsity levels (from 0 to 0.3 with intervals of 0.002) and calculate the area under the sparsity-accuracy curve.

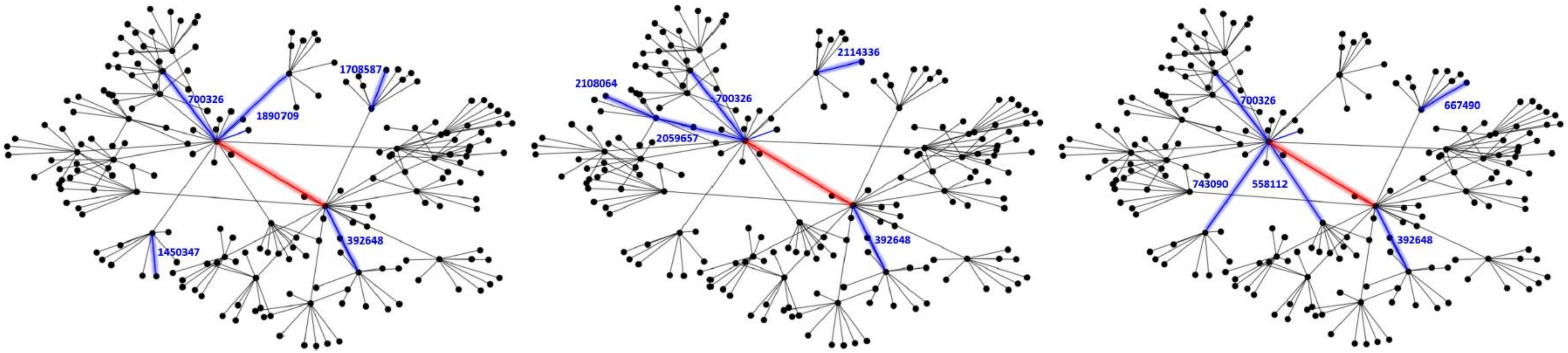
<u>Observation</u>

- We can observe that our model provides a higher quality of explanation for the predictions compared to other baselines.

EXPERIMENTS

- **Explanation Visualization**

Red : target event   Blue: candidate event



mean time interval = 1228523.4   mean time interval = 1475006.2   mean time interval = 612333.2

(a) GNNExplainer   **(b) PGExplainer**   **(c) TGIB**

➢ We marked the difference in occurrence timestamps between the target event and each of the five explanation events.

<u>Observation</u>

➢ The timestamps of the explanation events in TGIB are closer to the target events than GNNExplainer and PGExplainer.

➢ TGIB can capture temporal dependencies along with graph topology.

CONCLUSION

- We propose TGIB, a more reliable and practical explanation model for temporal graphs that can simultaneously perform prediction and explanation tasks.

- The main idea is to provide time-aware explanations based on Graph Information Bottleneck.
    - Restrict the flow of information to focus on the most relevant information for making accurate predictions.

- We demonstrate that our model shows significant performance in both prediction and explanation across various datasets.

Thank you!

[KDD' 24] Self-Explainable Temporal Graph Networks based on Graph Information Bottleneck
[Full Paper] https://arxiv.org/abs/2406.13214
[Source Code] https://github.com/sang-woo-seo/TGIB

[Email] sangwooseo@kaist.ac.kr