ADAPTIVE SEMANTIC EVOLUTION-DRIVEN REINFORCED DENOISING ON DYNAMIC GRAPHS

Anonymous authors
Paper under double-blind review

ABSTRACT

Dynamic graph learning relies on capturing complex topological and temporal evolutions. However, most models extract historical memory using a rigid exponential time-decay paradigm, struggling to model long-range dependencies and inducing severe extraction bias. This bias prevents models from reliably assessing incoming interactions, leading to indiscriminate aggregation that easily incorporates noisy links and propagates errors across nodes. To address this, we propose ATD-DyG, an adaptive evolution and reinforcement denoising framework. First, an adaptive semantic evolution kernel dynamically fits evolutionary patterns to reconstruct high-fidelity representations, eliminating temporal extraction bias. Using these representations as reliable decision priors, a reinforcement learning agent evaluates incoming interactions and proactively truncates high-risk noisy links. This prior-guided topological intervention shifts the paradigm from "passive absorption" to "proactive blocking", effectively halting noise propagation and enhancing model robustness. Experiments on 7 benchmark datasets demonstrate that ATD-DyG achieves advanced predictive performance and remarkable noise resistance.

1 INTRODUCTION

Dynamic graphs Zhang et al. (2023); Yang et al. (2024) have become the mainstream paradigm for characterizing the evolutionary laws of complex systems, such as social networks Jin et al. (2022), financial transactions Fan et al. (2024), and traffic flows Poursafaei et al. (2022). In recent years, continuous-time dynamic graph neural networks (CTDGs) Rossi et al. (2020); Xu et al. (2020b); Wang et al. (2021b) have demonstrated significant advantages owing to their fine-grained modeling capabilities for asynchronous event streams. However, in real-world scenarios accompanied by massive redundant interactions, accurately evaluating the semantic value of historical interactions and effectively resisting structural noise remain key challenges limiting model robustness.

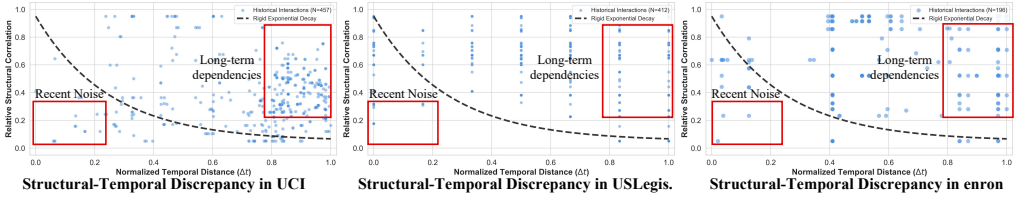


Figure 1: The structural correlation of historical neighbors (blue dots) for a randomly selected target node across normalized temporal distances significantly deviates from the exponential time-decay paradigm (dashed line), exposing the model's misestimation of recent noise (left box) and long-range dependencies (right box).

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

The key bottleneck restricting existing models from addressing this challenge is that they predominantly rely on the exponential time-decay paradigm to extract historical memory, which induces temporal modeling bias. Specifically, current mainstream temporal modeling architectures, including memory-based methods such as TGN Rossi et al. (2020) and JODIE Kumar et al. (2019), as well as neighborhood sequence-based methods such as DyGFormer Yu et al. (2023) and DyGMamba Yuan et al. (2025), utilize the monotonic exponential decay of physical time intervals as the primary basis for evaluating historical interactions. However, as illustrated in Fig. 1, the validity of interactions in real-world systems is predominantly determined by their intrinsic semantics, and a substantial number of high-value historical interactions do not strictly obey this rigid exponential time-decay paradigm. This mismatch between physical time and evolutionary semantics makes models tend to amplify recent local noise while forgetting key interactions with long-range dependencies. Consequently, models struggle to capture accurate evolutionary representations, failing to provide reliable prior references for evaluating new interactions.

This temporal extraction bias severely impairs the model's ability to discriminate incoming interactions. Lacking reliable evolutionary priors, existing architectures, such as TGN Rossi et al. (2020), TGAT Xu et al. (2020b), and TIDFormer Peng et al. (2025), typically degenerate into indiscriminate neighborhood aggregation when confronting new interactions. This blind process absorbs local noise from interacting neighbors, propagating errors across nodes and significantly undermining representation quality and model robustness.

To address these spatiotemporal modeling bottlenecks, we propose ATD-DyG, a robust representation framework for dynamic graphs based on adaptive evolution and reinforcement denoising. First, to eliminate temporal extraction bias, the framework introduces an adaptive semantic evolution kernel. By dynamically fitting non-local evolutionary laws, this mechanism breaks the limitations of monotonic decay, capturing multi-scale temporal dependencies and reconstructing high-fidelity evolutionary representations. Subsequently, utilizing these representations as reliable decision priors, we formulate dynamic denoising as a Markov Decision Process (MDP). Driven by the marginal log-likelihood gain of target links, a reinforcement learning agent evaluates incoming interactions within a global context and executes adaptive hard truncation on high-risk noisy links. This prior-guided topological intervention shifts the paradigm from "passive absorption" to "proactive blocking," fundamentally severing noise propagation paths at the topological level and ensuring model robustness. Our main contributions are summarized as follows:

- **Spatiotemporal Framework**: We propose ATD-DyG, combining an adaptive semantic evolution kernel with RL-driven topological denoising to reduce temporal extraction bias and noise propagation.
- **Proactive Denoising**: We formulate neighborhood selection as an MDP driven by marginal likelihood gain, using evolutionary priors to execute link truncation and transition from passive absorption to proactive blocking.
- **Experimental Evaluation**: Experiments across 7 benchmarks demonstrate the predictive performance and noise resilience of ATD-DyG under structural perturbations.

2 Related Work

2.1 Temporal Modeling for Dynamic Graphs

Temporal modeling for Continuous-Time Dynamic Graphs (CTDG) is primarily divided into two categories. Memory-based methods rely on specialized modules to maintain node states: TGN Rossi et al. (2020) utilizes RNN-driven memory networks to iteratively update node representations with asynchronous events, while NAT Luo & Li (2022) introduces a dictionary cache mechanism to achieve continuous state tracking. However, such iterative update mechanisms are prone to memory staleness, and implicit decay often biases

the model toward capturing recent transient dynamics, posing challenges in extracting long-term evolutionary trends. Neighborhood sequence-based methods enhance the temporal receptive field by directly processing historical interactions: for instance, GraphMixer Cong et al. (2023) utilizes Multi-Layer Perceptrons (MLP) to mix fixed-length interaction sequence features, while DyGFormer Yu et al. (2023) leverages Transformer architectures combined with time encoding to directly extract long-sequence correlations. Although these paradigms enhance modeling flexibility, they not only entail significant computational overhead $O(L^2)$ but also mostly treat timestamp encoding as a feature independent of semantics, making it difficult to flexibly coordinate long- and short-range temporal correlations according to the true value of interactions.

2.2 Structural Modeling for Dynamic Graphs

Structural modeling for dynamic graphs often utilizes time-aware attention such as TGAT Xu et al. (2020b) or contrastive learning such as TCL Wang et al. (2021a) for message aggregation. However, their indiscriminate global aggregation easily introduces outdated interactions, triggering the cascading diffusion of structural noise. Since soft filtering cannot physically block these contamination paths, proactive topological truncation is essential. This discrete decision-making process is naturally suited for Reinforcement Learning (RL), which bypasses the non-differentiability of discrete sampling Henderson et al. (2018); Nachum et al. (2018); Angermueller et al. (2019). While value-based methods Fan et al. (2020); Zou et al. (2019) and Actor-Critic methods Iqbal & Sha (2019); Haarnoja et al. (2018); Mnih et al. (2016) exist, policy-based RL Williams (1992); Rennie et al. (2017) exhibits advantages in directly optimizing truncation actions. Although GCPN You et al. (2018) and GDPNet Xue et al. (2021) validate RL in static graph denoising, assessing the long-term spatiotemporal impact of local truncation in continuous-time dynamic graphs (CTDGs) remains challenging. Consequently, we model neighborhood selection as a Markov Decision Process (MDP), driving an agent to proactively sever cascading contamination links and ensure robust representations.

3 Preliminary

Given a continuous-time dynamic graph G, each interaction is represented as (v_s, v_d, t), where v_s and v_d are the source and destination nodes. For a target interaction (v_s, v_d, t), we use the historical graph $G_{<t}$ and the candidate neighbor set $\mathcal{N}_{s\cup d}(t) = \mathcal{N}_s(t) \cup \mathcal{N}_d(t)$ for prediction. Since historical interactions exhibit non-uniform temporal relevance, outdated, redundant, or noisy neighbors may introduce structural interference into neighborhood aggregation. We therefore aim to retain an informative subset $\hat{\mathcal{N}}_{s\cup d}(t) \subseteq \mathcal{N}_{s\cup d}(t)$ that improves the prediction of the target interaction, formulated as

$$\hat{\mathcal{N}}_{s\cup d}(t) = \arg \max_{\mathcal{N}' \subseteq \mathcal{N}_{s\cup d}(t)} P\left(v_s, v_d, t \mid G_{<t}, \mathcal{N}'\right). \tag{1}$$

This formulation couples two challenges: historical interactions evolve with different temporal relevance, while neighborhood denoising requires discrete subset selection whose effect is reflected only through the downstream prediction. Accordingly, Section 4.1 models the temporal evolution and obtains time-aware neighbor representations, while Section 4.2 performs hard neighborhood selection to retain informative interactions and suppress structurally uninformative ones.

4 Methodology

This section introduces the ATD-DyG architecture illustrated in Figure 2, which comprises three core modules working synergistically: the Historical Interaction Extraction module retrieves contextual event sequences of the target interacting nodes; the Temporal Evolution and Aggregation module employs a fractional-order adaptive memory kernel and spatiotemporal attention to capture non-local temporal dependencies and perform

neighborhood feature fusion; and the Reinforcement Learning Neighborhood Selection module formalizes denoising as a Markov Decision Process (MDP), driving an agent to proactively sever noisy links to ultimately yield a robust graph topology.

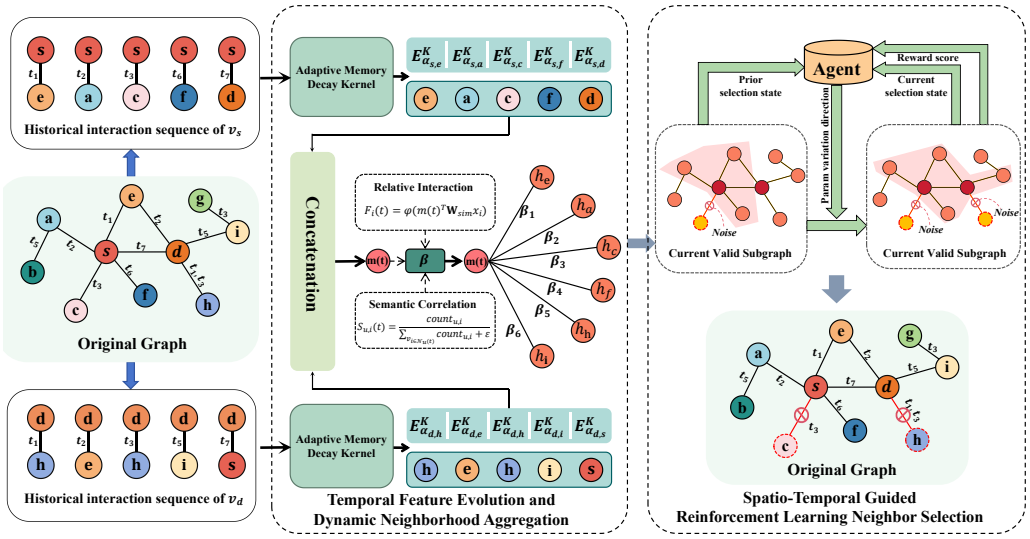


Figure 2: The overall architecture of ATD-DyG, where an adaptive semantic evolution kernel dynamically reconstructs high-fidelity representations from history (left), and an RL agent models neighbor selection as an MDP (right) to proactively truncate high-risk noisy links guided by spatiotemporal priors.

4.1 Adaptive Temporal Evolution and Dynamic Aggregation

To address the limitation where traditional exponential decay assumptions fail to characterize non-uniform evolutionary features in continuous-time dynamic graph modeling, this module synergizes a fractional-order adaptive memory kernel with a spatiotemporal-aware attention mechanism to adaptively assign weights to historical interactions across varying time intervals. This process effectively integrates local spatial structures with node semantic features, ultimately yielding intermediate state representations that characterize the feature intensity of neighbors, which serve as the feature input for the subsequent reinforcement learning neighbor selection module.

4.1.1 Adaptive Semantic Evolution Kernel

To overcome the exponential decay limitation, this module leverages the Mittag-Leffler operator in fractional calculus to capture non-local memory and long-range dependencies. By evaluating pairwise cosine similarity, it dynamically assigns fractional parameters to reshape historical weight distributions: low-order parameters induce a long-tail effect for similar neighbors, while high-order parameters accelerate forgetting for noisy neighbors (asymptotic decay proofs in Appendix A.5). For a time interval Δt, the kernel is defined as:

$$E^{K}_{\alpha_{s,i}}(-\lambda(\Delta t)^{\alpha_{s,i}}) = \sum_{k=0}^{K} \frac{(-\lambda(\Delta t)^{\alpha_{s,i}})^k}{\Gamma(\alpha_{s,i}k+1)} \tag{2}$$

where $\lambda > 0$ denotes base decay rate, $\Gamma(\cdot)$ is the Gamma function, and K controls the temporal perception boundary. To track the evolutionary trajectory of central node v_s, we aggregate its historical interaction set

$\mathcal{E}_s(t)$ into a contextual state $z_s(t) \in \mathbb{R}^{\frac{d_m}{2}}$:

$$z_s(t) = E_1^K(-\lambda(\Delta t_s))\mathbf{W}_{self}\mathbf{x}_s(t_{s,last}) + \sum_{(v_i,t_{s,i})\in\mathcal{E}_s(t)} E_{\alpha_{s,i}}^K(-\lambda\Delta t_{s,i}^{\alpha_{s,i}})\mathbf{W}_{neigh}\mathbf{x}_i(t_{s,i}) \tag{3}$$

where $\mathbf{x}_s(t_{s,last}), \mathbf{x}_i(t_{s,i}) \in \mathbb{R}^{d_n}$ denote node feature snapshots, $\mathbf{W}_{self}, \mathbf{W}_{neigh} \in \mathbb{R}^{\frac{d_m}{2}\times d_n}$ are projection matrices, and $\Delta t_s, \Delta t_{s,i}$ represent corresponding elapsed times. The fractional parameter $\alpha \in [0, 1)$ governing decay rate is determined by instantaneous semantic distance:

$$\alpha_{s,i} = 1 - \exp\left(-\frac{1-\cos(\mathbf{z}_s(t_{s,i}^-), \mathbf{x}_i(t_{s,i}))}{2\sigma_1^2}\right) \tag{4}$$

where $\mathbf{z}_s(t_{s,i}^-)$ tracks the dynamic semantic state of v_s prior to $t_{s,i}$, recursively updated upon neighbor aggregation; σ_1 is a smoothing factor. Higher semantic similarity yields a lower order to retain key interactions, whereas larger discrepancy pushes $\alpha \to 1$, recovering standard exponential decay to discard noise. The contextual state $z_d(t) \in \mathbb{R}^{\frac{d_m}{2}}$ for destination node v_d is computed analogously over $\mathcal{E}_d(t)$. Concatenating both states yields the global message $m(t) = z_s(t) \parallel z_d(t) \in \mathbb{R}^{d_m}$.

4.1.2 Dynamic Feature Refinement

To integrate structural, semantic, and temporal information, we first construct a joint relevance prior for each dynamic neighbor. For a candidate neighbor v_i of node v_u, where $v_u \in \{v_s, v_d\}$, the relative interaction weight is defined by local normalization:

$$S_{u,i}(t) = \frac{\mathrm{count}_{u,i}}{\sum_{v_j\in\mathcal{N}_u(t)} \mathrm{count}_{u,j} + \epsilon_s}, \tag{5}$$

where $\mathrm{count}_{u,i}$ denotes the cumulative interaction frequency between v_u and v_i, and ϵ_s is a smoothing constant. To capture semantic alignment, we measure the relevance between the candidate feature $\mathbf{x}_i \in \mathbb{R}^{d_n}$ and the current interaction message $\mathbf{m}(t)$ as

$$F_i(t) = \varphi\left(\mathbf{m}(t)^T\mathbf{W}_{\mathrm{sim}}\mathbf{x}_i\right), \tag{6}$$

where $\mathbf{W}_{\mathrm{sim}} \in \mathbb{R}^{d_m\times d_n}$ and $\varphi(\cdot)$ is the Sigmoid function. The structural and semantic signals are combined into the joint relevance prior

$$R_{u,i}(t) = S_{u,i}(t) \cdot F_i(t). \tag{7}$$

We then incorporate temporal evolution to obtain the time-aware attention weight:

$$\beta_i(t) = \frac{\exp\left(\sigma_2\left(E_{\alpha_{u,i}}^K\left(-\lambda(\Delta t_{u,i})^{\alpha_{u,i}}\right)\cdot \mathbf{x}_i^T\mathbf{W}_p\mathbf{m}(t)\cdot R_{u,i}(t)\right)\right)}{\sum_{j\in\mathcal{N}_{s\cup d}(t)}\exp\left(\sigma_2\left(E_{\alpha_{u,j}}^K\left(-\lambda(\Delta t_{u,j})^{\alpha_{u,j}}\right)\cdot \mathbf{x}_j^T\mathbf{W}_p\mathbf{m}(t)\cdot R_{u,j}(t)\right)\right)}, \tag{8}$$

where $\Delta t_{u,i} = t - t_{u,i}$, $\mathbf{W}_p \in \mathbb{R}^{d_n\times d_m}$, and σ_2 denotes the LeakyReLU activation. Finally, the refined intermediate representation of neighbor v_i is given by

$$h_i(t) = \beta_i(t)\cdot\mathbf{x}_i \in \mathbb{R}^{d_n}, \tag{9}$$

which is passed to the subsequent reinforcement learning module as the feature prior for topological pruning.

4.2 Reinforcement Learning-Based Topological Denoising

Although the adaptive semantic evolution kernel and spatiotemporal attention refine temporal interactions at the feature level and yield time-aware intermediate states $h_i(t)$, continuous attention weights cannot physically eliminate spurious links from the graph topology. To address this limitation, we utilize these time-aware representations and attention weights as evolutionary decision priors, formulate hard edge truncation as a finite-horizon sequential subset selection problem, and optimize it via policy-based reinforcement learning (RL) under a Markov Decision Process (MDP) framework, directly aligning local pruning decisions with the downstream link prediction objective.

Given the candidate set $\mathcal{N}_{s\cup d}(t)$ with cardinality $K = |\mathcal{N}_{s\cup d}(t)|$, candidate nodes are sorted in descending order of their attention weights $\beta_i(t)$ and evaluated sequentially. At decision step $i \in \{1, \ldots, K\}$, the environment state is defined as

$$S_i(t) = \left(i,\, v_i,\, \hat{\mathcal{N}}_{i-1}(t),\, c_s^{(i-1)}(t),\, c_d^{(i-1)}(t)\right), \tag{10}$$

where $\hat{\mathcal{N}}_0(t) = \emptyset, \hat{\mathcal{N}}_{i-1}(t)$ denotes the subset of neighbors selected up to step $i-1$, and $c_s^{(i-1)}(t)$, $c_d^{(i-1)}(t)$ denote the source and destination representations obtained by aggregating the neighbors retained up to step $i-1$. Upon executing a binary action $a_i \in \{0, 1\}$, the selected subset updates deterministically as $\hat{\mathcal{N}}_i(t) = \hat{\mathcal{N}}_{i-1}(t) \cup \{v_i \mid a_i = 1\}$. Given the predetermined evaluation schedule and deterministic state updates, future states and rewards depend strictly on $(S_i(t), a_i)$, rigorously satisfying the Markov property (see Appendix A.1 for formal proof and normalization).

The policy network encodes the current candidate along with the aggregation context induced by previously retained neighbors as $\phi_i(t) = x_i(t) \,\|\, h_i(t) \,\|\, c_s^{(i-1)}(t) \,\|\, c_d^{(i-1)}(t)$, and outputs a Bernoulli selection probability:

$$\pi_\theta(a_i \mid S_i(t)) = (p_i^\pi)^{a_i}(1 - p_i^\pi)^{1-a_i}, \quad p_i^\pi = \sigma\left(W_2\sigma_3(W_1\phi_i(t))\right), \tag{11}$$

where σ_3 denotes the ReLU activation and σ denotes the Sigmoid function.Notably, the attention-descending order serves solely as a deterministic evaluation schedule without restricting the feasible search space; every subset remains reachable through a valid sequence of binary decisions (see Appendix A.3 for proofs of subset reachability and terminal objective order-invariance).

For the downstream link prediction task, let $y \in \{0, 1\}$ be the ground-truth label of the target interaction. Given the selected subset $\hat{\mathcal{N}}_i(t)$, the predicted positive probability is $p_i = \sigma\left(W_{\text{link}}(c_s^{(i)}(t) \,\|\, c_d^{(i)}(t))\right)$, which induces a normalized Bernoulli likelihood $\mathbb{P}(y \mid S_i(t), a_i) = p_i^y(1 - p_i)^{1-y}$. We define the step-wise reward as the marginal improvement in the log-likelihood of the ground-truth target label:

$$r_i = \ell_i - \ell_{i-1}, \quad \text{with} \quad \ell_i = y \log p_i + (1 - y)\log(1 - p_i). \tag{12}$$

Consequently, conditioned on the previously retained subset, incorporating a candidate node yields a positive reward if and only if it enhances the predictive log-likelihood of the target label. For a complete decision trajectory $\tau = (a_1, \ldots, a_K)$, the cumulative reward forms a telescoping sum:

$$R(\tau) = \sum_{i=1}^{K} r_i = \sum_{i=1}^{K}(\ell_i - \ell_{i-1}) = \ell_K - \ell_0 = \mathcal{L}_{\text{BCE}}^{(0)} - \mathcal{L}_{\text{BCE}}^{(K)}, \tag{13}$$

where $\mathcal{L}_{\text{BCE}}^{(0)}$ denotes the initial prediction loss prior to any candidate neighborhood aggregation (relying solely on intrinsic self-evolution features), which is independent of the selection policy. Therefore, maximizing the expected cumulative reward is mathematically equivalent to minimizing the final Binary Cross-Entropy loss: $\max_\theta \mathbb{E}_{\tau\sim\pi_\theta}[R(\tau)] \Longleftrightarrow \min_\theta \mathbb{E}_{\tau\sim\pi_\theta}[\mathcal{L}_{\text{BCE}}^{(K)}]$.(Theorem in Appendix A.2 provides the formal proof of this reward-objective alignment).

Finally, we optimize the policy parameters using Self-Critical Training (SCT) to reduce the variance of policy gradient estimation, setting the discount factor $\gamma = 1$ to maintain the exact objective equivalence. Let $R_{\text{sample},i}$ and $R_{\text{greedy},i}$ denote the returns-to-go from step i under a sampled exploration trajectory and a greedily decoded baseline trajectory, respectively. The policy gradient estimator is given by

$$\widehat{\nabla_\theta J(\theta)} = \frac{1}{K}\mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{i=1}^{K}\nabla_\theta \log \pi_\theta(a_i \mid S_i(t)) \cdot (R_{\text{sample},i} - R_{\text{greedy},i})\right]. \tag{14}$$

The greedy baseline, being independent of the sampled action, preserves gradient unbiasedness while effectively suppressing variance(formal proof in Appendix A.4).. Policy parameters are then updated via $\theta \leftarrow \theta + \eta\widehat{\nabla_\theta J(\theta)}$, where η is the learning rate.

5 EXPERIMENTS

We evaluate ATD-DyG against 11 state-of-the-art CTDG baselines across 7 benchmark datasets. Detailed dataset and baseline descriptions are provided in Appendix C.

5.1 OVERALL PERFORMANCE ANALYSIS FOR DYNAMIC LINK PREDICTION

Table 1 presents the AUC-ROC results across seven benchmark datasets under both transductive and inductive settings. ATD-DyG achieves strong overall performance, obtaining average ranks of 1.1 and 1.3 in the transductive and inductive settings, respectively. While competing methods remain competitive on several individual datasets, ATD-DyG consistently maintains a high ranking across different datasets and evaluation settings. These results demonstrate the effectiveness of combining the Adaptive Semantic Evolution Kernel for modeling long-range temporal dependencies with the reinforcement denoising mechanism for selectively filtering potentially noisy interactions.

Table 1: The AUC-ROC metrics of the seven datasets under both transductive and inductive settings.

Datasets	Setting	JODIE	DyRep	TGAT	TGN	CAWN	EdgeBank	TCL	AdaDy-GNN	GraphMixer	DyGFormer	DyG-Mamba	ATD-DyG
UCI	Transductive	90.44±0.49	68.77±2.34	78.53±0.74	92.03±1.13	93.87±1.08	77.30±0.00	87.82±1.36	92.59±0.12	91.81±0.67	94.49±0.26	93.51±0.24	**95.43±0.18**
	Inductive	78.80±0.94	58.08±1.81	77.64±0.38	86.68±2.29	90.40±0.11	-	84.49±1.82	82.04±0.02	89.30±0.57	92.63±0.13	91.99±0.03	**93.74±0.11**
Wikipedia	Transductive	96.33±0.07	94.37±0.09	96.67±0.07	98.37±0.07	98.54±0.04	90.78±0.00	95.84±0.18	98.80±0.00	96.62±0.03	98.91±0.02	98.45±0.12	**99.21±0.13**
	Inductive	94.33±0.27	91.49±0.45	95.90±0.09	97.72±0.03	98.03±0.04	-	95.57±0.20	98.37±0.01	96.30±0.04	98.48±0.03	98.57±0.03	**99.03±0.08**
LastFM	Transductive	70.49±1.66	71.16±1.89	71.59±0.18	78.47±2.94	85.92±0.10	83.77±0.00	64.06±1.16	93.35±1.02	73.53±0.12	93.05±0.10	94.71±0.23	**96.16±0.04**
	Inductive	81.13±3.39	82.24±1.51	76.99±0.29	82.61±3.15	87.82±0.12	-	70.84±0.85	94.52±1.24	80.37±0.18	94.08±0.16	94.83±0.01	**96.47±0.07**
Enron	Transductive	87.96±0.52	84.89±3.00	68.89±1.10	88.32±0.99	90.45±0.14	87.06±0.00	75.74±0.72	81.48±0.24	84.38±0.21	**93.33±0.13**	93.13±0.68	93.15±0.16
	Inductive	81.96±1.34	76.34±4.20	64.63±1.74	78.83±1.11	87.02±0.50	-	72.33±0.99	78.91±0.41	76.51±0.71	90.69±0.26	**90.84±0.18**	90.38±0.17
USLegis.	Transductive	82.85±1.07	82.28±0.32	75.84±1.99	83.34±0.43	77.16±0.39	62.57±0.00	76.27±0.63	81.82±0.76	76.96±0.79	77.90±0.58	80.11±0.32	**91.89±0.61**
	Inductive	58.12±2.35	61.07±0.56	48.27±3.50	62.38±0.48	51.49±1.13	-	50.43±1.48	64.49±0.87	47.20±0.89	53.21±3.04	56.56±1.08	**77.96±0.59**
mooc	Transductive	83.81±2.09	85.03±0.58	87.11±0.19	91.21±1.15	80.38±0.26	78.86±0.00	83.12±0.18	86.82±0.38	84.01±0.17	87.91±0.58	86.46±0.14	**91.98±0.17**
	Inductive	82.75±0.83	81.06±0.94	80.81±0.67	88.00±1.80	71.57±1.07	-	72.09±0.56	89.31±0.27	84.01±0.17	87.04±0.35	91.05±1.02	**93.13±0.11**
UNtrade	Transductive	62.28±0.50	58.82±0.98	62.72±0.12	59.99±3.05	67.05±0.21	52.11±0.00	63.76±0.07	69.26±0.33	63.48±0.37	67.25±1.05	69.22±0.52	**78.66±1.12**
	Inductive	58.15±0.33	54.06±0.55	51.61±3.61	49.10±2.14	60.63±2.81	-	59.41±0.72	66.53±0.29	58.39±0.18	63.83±0.12	66.78±0.80	**76.38±0.18**
Avg.Rank	Transductive	7.9	9.0	8.7	5.4	5.7	10.3	9.7	4.7	8.0	3.3	4.1	**1.1**
	Inductive	7.4	8.4	9.7	6.1	6.0	-	8.9	4.1	7.6	4.0	2.6	**1.3**

5.2 CROSS-SCALE REINFORCEMENT DENOISING ANALYSIS

To verify if the agent transcends traditional time-decay bias, we analyze neighbor retention across different temporal windows (Fig. 3). On MOOC and UCI datasets, the *distant window* maintains a high retention rate (e.g., 39.80% in MOOC), effectively breaking the monotonic decay trend. The scatter plots further reveal that many long-range interactions still yield selection probabilities near 1.0. This confirms that ATD-DyG prioritizes intrinsic semantic and topological value over mere recency, validating its capability to capture non-local long-range dependencies through active denoising.

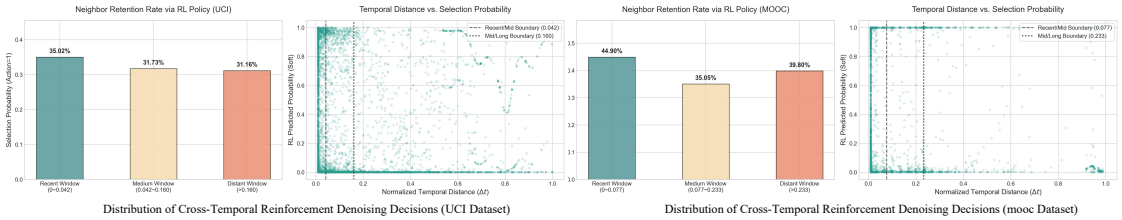


Figure 3: Cross-temporal reinforcement denoising decision distributions across multiple datasets

5.3 Dynamic Link Prediction Robustness under Structural Noise

To evaluate model robustness, we inject structural noise ($\rho \in [0, 0.3]$) into the Enron dataset, adopting Mean Reciprocal Rank (MRR) and the Decrease rate (DEC) for evaluation. As shown in Table 2, baselines lacking proactive denoising degrade severely under extreme noise ($\rho = 0.3$), with DyRep and TGN exhibiting DECs up to 51.2% and 43.4%, respectively. In contrast, ATD-DyG keeps its DEC within 7.8%. Notably, its absolute performance under severe noise (MRR = 0.1494) still surpasses the best noise-free baseline. This confirms that our framework effectively truncates spurious links, delivering strong resilience against structural perturbations.

Table 2: Robustness evaluation for dynamic link prediction under structural noise.

Models	Settings	$\rho = 0$	$\rho = 0.01$		$\rho = 0.03$		$\rho = 0.1$		$\rho = 0.3$	
		MRR	MRR	DEC	MRR	DEC	MRR	DEC	MRR	DEC
DyRep	Transductive	0.0312	0.0311	0.3%	0.0305	2.2%	0.0268	14.1%	0.0152	51.2%
	Inductive	0.0165	0.0164	0.6%	0.0164	0.6%	0.0161	2.4%	0.0126	23.6%
TGN	Transductive	0.0159	0.0158	0.6%	0.0156	1.9%	0.0141	11.3%	0.0111	30.2%
	Inductive	0.0143	0.0142	0.7%	0.0140	2.1%	0.0116	18.9%	0.0081	43.4%
Ada-DyGNN	Transductive	0.0989	0.0988	0.1%	0.0981	0.8%	0.0958	3.1%	0.0941	5.0%
	Inductive	0.0886	0.0885	0.1%	0.0884	0.2%	0.0878	0.9%	0.0855	3.5%
ATD-DyG	Transductive	0.1613	0.1612	0.1%	0.1603	0.6%	0.1597	0.9%	0.1494	7.3%
	Inductive	0.1609	0.1608	0.1%	0.1593	0.9%	0.1574	2.1%	0.1483	7.8%

5.4 Ablation Analysis

To verify core components, we conduct ablation studies on UCI, Enron, and LastFM (Table 3). The full ATD-DyG consistently achieves optimal performance. Replacing the Mittag-Leffler function with traditional exponential decay causes a significant accuracy drop, proving the adaptive semantic evolution kernel's crucial role in overcoming monotonic decay to capture non-local temporal dependencies. Additionally, three variants lacking the RL module (Select-All, Select-None, Select-Random) suffer severe degradation, with the random strategy dropping most. This proves that indiscriminate aggregation easily introduces structural noise, whereas RL-based active discrimination precisely truncates polluted links, ensuring representation robustness.

Table 3: Ablation study on three datasets (AUC %).

Methods	UCI		Enron		LastFM	
	Trans.	Ind.	Trans.	Ind.	Trans.	Ind.
ATD-DyG (Full)	**95.43**	**93.74**	**93.15**	**90.38**	**97.15**	**96.47**
ATD-DyG-FCHIF-Exp	94.19	92.58	91.82	87.12	94.86	95.01
ATD-DyG-RL-Select-All	94.38	93.49	90.23	87.73	94.52	94.08
ATD-DyG-RL-Select-None	94.02	92.11	89.81	86.84	93.79	93.79
ATD-DyG-RL-Select-Random	92.21	90.83	88.79	85.19	91.88	92.32

5.5 Comparison of Training Time and Model Parameters

Figure 4 presents a comparison of training time and parameter size on the UCI, Enron, and LastFM datasets. Models omitted from the figure encountered Out-of-Memory (OOM) errors. It is evident that ATD-DyG achieves the optimal performance while maintaining relatively low temporal and memory overhead.

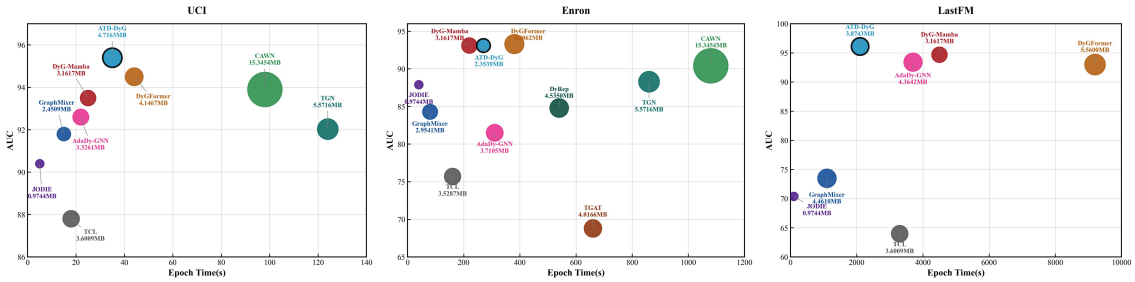


Figure 4: Comparison of performance, training efficiency, and model complexity.

6 Conclusion and Future Work

This paper introduces a continuous-time dynamic graph learning framework that resolves spatiotemporal mismatch and cascading noise diffusion. We propose an adaptive semantic evolution kernel to fit non-local evolutionary laws and accurately extract cross-temporal long-range dependencies. By integrating a reinforcement learning-based topological denoising mechanism into the aggregation process, we enable the proactive truncation of spurious links, improving model robustness and achieving state-of-the-art results on benchmark datasets. While highly effective, our framework is sensitive to the trade-off between memory retention and noise filtering, with higher fractional truncation orders introducing significant computational overhead. In the future, potential areas for improvement in our framework include extending the adaptive kernel to heterogeneous networks, optimizing parameter sensitivity, and exploring continuous action spaces for fine-grained edge modulation.

References

Christof Angermueller, David Dohan, David Belanger, Ramya Deshpande, Kevin Murphy, and Lucy Colwell. Model-based reinforcement learning for biological sequence design. In *International Conference on Learning Representations*, 2019.

Weilin Cong, Si Zhang, Jian Kang, Baichuan Yuan, Hao Wu, Xin Zhou, Hanghang Tong, and Mehrdad Mahdavi. Do we really need complicated model architectures for temporal networks? *arXiv preprint arXiv:2302.11636*, 2023.

Jianqing Fan, Zhaoran Wang, Yuchen Xie, and Zhuoran Yang. A theoretical analysis of deep Q-learning. In *Proceedings of the Conference on Learning for Dynamics and Control*, pp. 486–489. PMLR, 2020.

Yan Fan, Yu Wang, Pengfei Zhu, and Qinghua Hu. Dynamic sub-graph distillation for robust semi-supervised continual learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 11927–11935, 2024.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International Conference on Machine Learning*, pp. 1861–1870. PMLR, 2018.

Peter Henderson, Riashat Islam, Philip Bachman, Joelle Pineau, Doina Precup, and David Meger. Deep reinforcement learning that matters. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Shariq Iqbal and Fei Sha. Actor-attention-critic for multi-agent reinforcement learning. In *International Conference on Machine Learning*, pp. 2961–2970. PMLR, 2019.

Ming Jin, Yuan-Fang Li, and Shirui Pan. Neural temporal walks: Motif-aware representation learning on continuous-time dynamic graphs. *Advances in Neural Information Processing Systems*, 35:19874–19886, 2022.

Srijan Kumar, Xikun Zhang, and Jure Leskovec. Predicting dynamic embedding trajectory in temporal interaction networks. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1269–1278, 2019.

Hanjie Li, Changsheng Li, Kaituo Feng, Ye Yuan, Guoren Wang, and Hongyuan Zha. Robust knowledge adaptation for dynamic graph neural networks. *IEEE Transactions on Knowledge and Data Engineering*, 36(11):6920–6933, 2024.

Yuhong Luo and Pan Li. Neighborhood-aware scalable temporal network representation learning. In *Proceedings of the Learning on Graphs Conference*, pp. 1–1. PMLR, 2022.

Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *International Conference on Machine Learning*, pp. 1928–1937. PMLR, 2016.

Ofir Nachum, Shixiang Shane Gu, Honglak Lee, and Sergey Levine. Data-efficient hierarchical reinforcement learning. *Advances in Neural Information Processing Systems*, 31, 2018.

Jie Peng, Zhewei Wei, and Yuhang Ye. TIDFormer: Exploiting temporal and interactive dynamics makes a great dynamic graph transformer. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 2245–2256, 2025.

Farimah Poursafaei, Shenyang Huang, Kellin Pelrine, and Reihaneh Rabbany. Towards better evaluation for dynamic link prediction. *Advances in Neural Information Processing Systems*, 35:32928–32941, 2022.

Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. Self-critical sequence training for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7008–7024, 2017.

Emanuele Rossi, Ben Chamberlain, Fabrizio Frasca, Davide Eynard, Federico Monti, and Michael Bronstein. Temporal graph networks for deep learning on dynamic graphs. *arXiv preprint arXiv:2006.10637*, 2020.

Rakshit Trivedi, Mehrdad Farajtabar, Prasenjeet Biswal, and Hongyuan Zha. DyRep: Learning representations over dynamic graphs. In *International Conference on Learning Representations*, 2019.

Lu Wang, Xiaofu Chang, Shuang Li, Yunfei Chu, Hui Li, Wei Zhang, Xiaofeng He, Le Song, Jingren Zhou, and Hongxia Yang. TCL: Transformer-based dynamic graph modelling via contrastive learning. *arXiv preprint arXiv:2105.07944*, 2021a.

Xuhong Wang, Ding Lyu, Mengjian Li, Yang Xia, Qi Yang, Xinwen Wang, Xinguang Wang, Ping Cui, Yupu Yang, Bowen Sun, et al. APAN: Asynchronous propagation attention network for real-time temporal graph embedding. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pp. 2628–2638, 2021b.

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3):229–256, 1992.

Da Xu, Chuanwei Ruan, Evren Korpeoglu, Sushant Kumar, and Kannan Achan. Inductive representation learning on temporal graphs. *arXiv preprint arXiv:2002.07962*, 2020a.

Da Xu, Chuanwei Ruan, Evren Korpeoglu, Sushant Kumar, and Kannan Achan. Inductive representation learning on temporal graphs. *arXiv preprint arXiv:2002.07962*, 2020b.

Fuzhao Xue, Aixin Sun, Hao Zhang, and Eng Siong Chng. GDPNet: Refining latent multi-view graph for relation extraction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 14194–14202, 2021.

Kuo Yang, Zhengyang Zhou, Qihe Huang, Limin Li, Yuxuan Liang, and Yang Wang. Improving generalization of dynamic graph learning via environment prompt. *Advances in Neural Information Processing Systems*, 37:70048–70075, 2024.

Jiaxuan You, Bowen Liu, Zhitao Ying, Vijay Pande, and Jure Leskovec. Graph convolutional policy network for goal-directed molecular graph generation. *Advances in Neural Information Processing Systems*, 31, 2018.

Le Yu, Leilei Sun, Bowen Du, and Weifeng Lv. Towards better dynamic graph learning: New architecture and unified library. *Advances in Neural Information Processing Systems*, 36:67686–67700, 2023.

Haonan Yuan, Qingyun Sun, Zhaonan Wang, Xingcheng Fu, Cheng Ji, Yongjian Wang, Bo Jin, and Jianxin Li. DG-Mamba: Robust and efficient dynamic graph structure learning with selective state space models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 22272–22280, 2025.

Yao Zhang, Yun Xiong, Yongxiang Liao, Yiheng Sun, Yucheng Jin, Xuehao Zheng, and Yangyong Zhu. TIGER: Temporal interaction graph embedding with restarts. In *Proceedings of the ACM Web Conference*, pp. 478–488, 2023.

Shaofeng Zou, Tengyu Xu, and Yingbin Liang. Finite-sample analysis for SARSA with linear function approximation. *Advances in Neural Information Processing Systems*, 32, 2019.

A Theoretical Proofs and Additional Details

This appendix provides the theoretical justifications for the reinforcement learning-based topological denoising framework in Section 4.2 and the adaptive temporal evolution kernel in Section 4.1. Specifically, we establish five key theoretical properties: (i) the sequential neighborhood-selection process satisfies the Markov property and induces a valid probability measure; (ii) the proposed marginal log-likelihood reward is strictly aligned with the downstream link prediction objective; (iii) the attention-based evaluation order preserves candidate subset reachability and the terminal cumulative objective; (iv) the self-critical baseline maintains an unbiased expected policy gradient while reducing estimation variance; and (v) the adaptive semantic evolution kernel smoothly transitions between long-tail memory retention for similar neighbors and fast forgetting for noisy neighbors.

A.1 Markov Property and Probability Measure

We first establish that the sequential neighborhood-selection process defined in Section 4.2 constitutes a finite-horizon Markov decision process. Recall that the environment state at decision step i is defined as

$$S_i(t) = \left(i,\, v_i,\, \hat{\mathcal{N}}_{i-1}(t),\, c_s^{(i-1)}(t),\, c_d^{(i-1)}(t)\right), \tag{15}$$

where $\hat{\mathcal{N}}_{i-1}(t)$ denotes the subset selected before step i, and $c_s^{(i-1)}(t)$ and $c_d^{(i-1)}(t)$ denote the corresponding aggregation states of the source and destination nodes.

Proposition 1 (Markov Property). Given the current state $S_i(t)$ and action a_i, the next state and immediate reward are conditionally independent of all preceding states and actions:

$$\begin{aligned}&\mathbb{P}\left(S_{i+1}(t), r_i \mid S_1, a_1, \ldots, S_i, a_i\right)\\ &\quad = \mathbb{P}\left(S_{i+1}(t), r_i \mid S_i(t), a_i\right).\end{aligned} \tag{16}$$

Proof. At step i, the binary action a_i determines whether the current candidate v_i is retained. Specifically,

$$\hat{\mathcal{N}}_i(t) = \hat{\mathcal{N}}_{i-1}(t) \cup \{v_i \mid a_i = 1\}. \tag{17}$$

The aggregation states are then updated from the current state and action:

$$\left(c_s^{(i)}(t), c_d^{(i)}(t)\right) = F_i\left(S_i(t), a_i\right), \tag{18}$$

where F_i denotes the deterministic aggregation update induced by the model architecture. Since the candidate ordering is predetermined, the next candidate v_{i+1} is determined by the current decision step. Therefore, the next state is determined by the current state and action.

Moreover, the immediate reward is computed from the predictive likelihood before and after the current selection decision. Hence, it is also determined by the current state and action. No additional information from earlier states or actions is required once $S_i(t)$ is given, which establishes the Markov property in Eq. equation 16. □

The policy defined in Eq. (12) is a Bernoulli distribution over the binary action space. Its normalization follows directly from

$$\sum_{a_i \in \{0,1\}} \pi_\theta(a_i \mid S_i(t)) = p_i^\pi + (1 - p_i^\pi) = 1. \tag{19}$$

Therefore, for a trajectory $\tau = (a_1, \ldots, a_K)$, the induced trajectory probability is

$$\mathbb{P}_\theta(\tau) = \prod_{i=1}^{K} \pi_\theta(a_i \mid S_i(t)). \tag{20}$$

Since the trajectory space is finite, $\tau \in \{0,1\}^K$, the induced probability measure is normalized:

$$\sum_{\tau \in \{0,1\}^K} \mathbb{P}_\theta(\tau) = 1. \tag{21}$$

Thus, the proposed sequential neighborhood-selection process defines a valid probability measure over all possible selection trajectories.

A.2 Reward-Objective Alignment

We next establish that the proposed reward is directly connected to the downstream link prediction objective, rather than treating the absolute log-likelihood as an arbitrary reward.

Let $y \in \{0,1\}$ denote the ground-truth label of the target interaction. After step i, the link predictor produces

$$p_i = \sigma\left(W_{\text{link}}\left(c_s^{(i)}(t) \| c_d^{(i)}(t)\right)\right). \tag{22}$$

The corresponding conditional distribution of the ground-truth label is the normalized Bernoulli distribution

$$\mathbb{P}\left(y \mid S_i(t), a_i\right) = p_i^y (1-p_i)^{1-y}. \tag{23}$$

Indeed,

$$\sum_{y \in \{0,1\}} p_i^y (1-p_i)^{1-y} = p_i + (1-p_i) = 1. \tag{24}$$

The ground-truth log-likelihood is defined as

$$\ell_i = \log \mathbb{P}\left(y \mid S_i(t), a_i\right) = y \log p_i + (1-y)\log(1-p_i). \tag{25}$$

Rather than using the absolute log-likelihood as the reward, we define the step-wise reward as its marginal improvement:

$$r_i = \ell_i - \ell_{i-1}. \tag{26}$$

Therefore, r_i measures the change in the downstream predictive objective caused by the current neighborhood-selection decision.

Theorem 1 (Equivalence Between Cumulative Reward and BCE Reduction). For any selection trajectory $\tau = (a_1, \ldots, a_K)$, the cumulative reward is exactly equal to the reduction in the downstream BCE loss.

Proof. By the definition of the step-wise reward,

$$\begin{aligned} R(\tau) &= \sum_{i=1}^{K} r_i = \sum_{i=1}^{K} (\ell_i - \ell_{i-1}) \\ &= \ell_K - \ell_0. \end{aligned} \tag{27}$$

For binary link prediction, the BCE loss at step i is

$$\mathcal{L}_{\text{BCE}}^{(i)} = -y \log p_i - (1-y)\log(1-p_i) = -\ell_i. \tag{28}$$

Substituting Eq. equation 28 into Eq. equation 27 gives

$$R(\tau) = \mathcal{L}_{\mathrm{BCE}}^{(0)} - \mathcal{L}_{\mathrm{BCE}}^{(K)}. \tag{29}$$

The initial loss $\mathcal{L}_{\mathrm{BCE}}^{(0)}$ is evaluated before neighborhood selection and is independent of the policy parameters θ. Consequently,

$$\arg\max_{\theta} \mathbb{E}_{\tau\sim\pi_\theta}\left[R(\tau)\right] = \arg\min_{\theta} \mathbb{E}_{\tau\sim\pi_\theta}\left[\mathcal{L}_{\mathrm{BCE}}^{(K)}\right]. \tag{30}$$

Hence, maximizing the expected cumulative reward is mathematically equivalent to minimizing the expected final BCE loss. □

A.3 Attention Ordering and Subset Reachability

The candidates in Section 4.2 are evaluated in descending order of their attention weights. We next clarify the role of this ordering. In particular, we show that the deterministic ordering does not restrict the feasible subset space.

Proposition 2 (Reachability of Every Candidate Subset). Let $(v_1, \ldots, v_K)$ denote the fixed candidate ordering. For any subset $\mathcal{A} \subseteq \mathcal{N}_{s\cup d}(t)$, there exists a binary action sequence that produces $\hat{\mathcal{N}}_K(t) = \mathcal{A}$.

Proof. For each candidate v_i, define

$$a_i = \begin{cases} 1, & v_i \in \mathcal{A}, \\ 0, & v_i \notin \mathcal{A}. \end{cases} \tag{31}$$

Using the subset update in Eq. equation 17, the resulting final subset is

$$\hat{\mathcal{N}}_K(t) = \{v_i \mid a_i = 1\} = \mathcal{A}. \tag{32}$$

Thus, every subset of the candidate set remains reachable under the fixed attention-based evaluation order. The ordering therefore determines only the sequence in which candidates are evaluated and does not remove any candidate subset from the feasible action space. □

Theorem 2 (Order Invariance of the Cumulative Objective). Let $\mathcal{A}$ be a fixed selected subset and let ρ denote any permutation of its elements. Let $\mathcal{A}_i^{(\rho)}$ denote the subset containing the first i elements of $\mathcal{A}$ under ordering ρ. For a fixed final selected subset, the cumulative marginal log-likelihood improvement is

$$\begin{aligned} &\sum_{i=1}^{|\mathcal{A}|} \left[\ell\left(\mathcal{A}_i^{(\rho)}\right) - \ell\left(\mathcal{A}_{i-1}^{(\rho)}\right)\right] \\ &\qquad = \ell(\mathcal{A}) - \ell(\emptyset), \end{aligned} \tag{33}$$

and hence depends only on the initial and final selected subsets.

Proof. The left-hand side telescopes:

$$\begin{aligned} &\sum_{i=1}^{|\mathcal{A}|} \left[\ell\left(\mathcal{A}_i^{(\rho)}\right) - \ell\left(\mathcal{A}_{i-1}^{(\rho)}\right)\right] \\ &\qquad = \ell\left(\mathcal{A}_{|\mathcal{A}|}^{(\rho)}\right) - \ell\left(\mathcal{A}_0^{(\rho)}\right) \\ &\qquad = \ell(\mathcal{A}) - \ell(\emptyset). \end{aligned} \tag{34}$$

For the same final selected subset, the terminal prediction objective is therefore the same regardless of the order in which its elements are evaluated. Hence, the cumulative objective associated with a fixed selected subset is invariant to the evaluation order. □

The above result does not imply that the intermediate states or policy probabilities are identical under different evaluation orders. The order may affect the intermediate trajectory and the factorization of the policy over decision steps. However, every candidate subset remains reachable, and the cumulative marginal objective for a fixed final subset is unchanged. Therefore, the attention-based ordering acts as a deterministic evaluation schedule rather than a constraint on the feasible neighborhood-selection solutions.

A.4 Self-Critical Policy Gradient

Let the expected cumulative reward be

$$J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta}\left[R(\tau)\right]. \tag{35}$$

Since the reward defined in Eq. equation 26 telescopes exactly to the final BCE reduction, we use an undiscounted objective with $\gamma = 1$.

For a trajectory $\tau = (a_1, \ldots, a_K)$, its log-probability is

$$\log \mathbb{P}_\theta(\tau) = \sum_{i=1}^{K} \log \pi_\theta(a_i \mid S_i). \tag{36}$$

The likelihood-ratio identity gives the policy gradient

$$\nabla_\theta J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{i=1}^{K} \nabla_\theta \log \pi_\theta(a_i \mid S_i) R_{\text{sample},i}\right], \tag{37}$$

where $R_{\text{sample},i}$ denotes the return associated with the sampled trajectory from decision step i onward.

Proposition 3 (Unbiasedness of the Self-Critical Baseline). Let $R_{\text{greedy},i}$ denote the return obtained by greedily decoding the remaining decisions from state S_i. Conditioned on S_i, the greedy baseline is independent of the sampled action a_i and is treated as a stop-gradient baseline with respect to θ. Then

$$\mathbb{E}_{a_i\sim\pi_\theta(\cdot|S_i)}\left[\nabla_\theta \log \pi_\theta(a_i \mid S_i) R_{\text{greedy},i}\right] = 0. \tag{38}$$

Consequently, subtracting the greedy return preserves the expected policy gradient:

$$\begin{aligned}\nabla_\theta J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta}\Bigg[\sum_{i=1}^{K} &\nabla_\theta \log \pi_\theta(a_i \mid S_i) \\ &\cdot (R_{\text{sample},i} - R_{\text{greedy},i})\Bigg].\end{aligned} \tag{39}$$

Proof. Conditioned on S_i, the greedy return is fixed with respect to the sampled action. Therefore,

$$\begin{aligned}&\mathbb{E}_{a_i}\left[\nabla_\theta \log \pi_\theta(a_i \mid S_i) R_{\text{greedy},i}\right] \\ &\quad= R_{\text{greedy},i} \sum_{a_i} \pi_\theta(a_i \mid S_i) \nabla_\theta \log \pi_\theta(a_i \mid S_i) \\ &\quad= R_{\text{greedy},i} \sum_{a_i} \nabla_\theta \pi_\theta(a_i \mid S_i) \\ &\quad= R_{\text{greedy},i} \nabla_\theta \sum_{a_i} \pi_\theta(a_i \mid S_i) = 0.\end{aligned} \tag{40}$$

Thus, the self-critical baseline does not change the expectation of the policy-gradient estimator while providing a state-dependent reference for reducing its variance. □

A.5 Asymptotic Behavior of the Adaptive Semantic Evolution Kernel

This section establishes the asymptotic behavior of the adaptive semantic evolution kernel $E^K_{\alpha_{s,i}}(-\lambda(\Delta t)^{\alpha_{s,i}})$, showing that it provides a continuous transition between exponential forgetting and long-tail power-law memory according to the semantic discrepancy between the central node v_s and its neighbor v_i.

Proof of the Exponential Decay Boundary. The fractional-order parameter is defined as

$$\alpha_{s,i} = 1 - \exp\left(-\frac{1-\cos(z_s, x_i)}{2\sigma_1^2}\right). \tag{41}$$

When the semantic feature difference between the central node v_s and neighbor v_i is large, i.e., $\cos(z_s, x_i) \to -1$, we have $1 - \cos(z_s, x_i) \to 2$, and hence

$$\alpha_{s,i} \to 1 - \exp\left(-\frac{1}{\sigma_1^2}\right). \tag{42}$$

Thus, for sufficiently small σ_1, the fractional order approaches the exponential-order boundary, $\alpha_{s,i} \approx 1$. In the limiting case $\alpha_{s,i} = 1$, and as the truncation order $K \to \infty$, the Mittag–Leffler series reduces to

$$\begin{aligned} E_1(-\lambda\Delta t) &= \sum_{k=0}^{\infty} \frac{(-\lambda\Delta t)^k}{\Gamma(k+1)} \\ &= \sum_{k=0}^{\infty} \frac{(-\lambda\Delta t)^k}{k!} = e^{-\lambda\Delta t}. \end{aligned} \tag{43}$$

Therefore, for semantically dissimilar or weakly relevant neighbors, the adaptive kernel approaches the conventional exponential-decay boundary, corresponding to rapid forgetting of historical interactions. **Proof of Power-law Decay for Long-tail Memory.** For semantically similar neighbors, the semantic discrepancy is small, which yields a relatively small fractional order $\alpha_{s,i} \in (0, 1)$. As the semantic similarity increases, $\alpha_{s,i}$ approaches zero, corresponding to a progressively slower decay regime. For any fixed $\alpha \in (0, 1)$, consider the standard two-parameter Mittag–Leffler function $E_{\alpha,\beta}(z)$. For $\Delta t \to \infty$ and $z = -\lambda(\Delta t)^\alpha \to -\infty$, its asymptotic expansion along the negative real axis is

$$E_{\alpha,1}\left(-\lambda(\Delta t)^\alpha\right) = -\sum_{k=1}^{N} \frac{(-\lambda(\Delta t)^\alpha)^{-k}}{\Gamma(1-\alpha k)} + O\left(|\lambda(\Delta t)^\alpha|^{-N-1}\right). \tag{44}$$

Taking the leading term with $k = 1$ gives

$$E_{\alpha,1}\left(-\lambda(\Delta t)^\alpha\right) \sim \frac{1}{\lambda\Gamma(1-\alpha)} \Delta t^{-\alpha}. \tag{45}$$

Hence, for $0 < \alpha < 1$, the kernel exhibits power-law decay, $E_{\alpha,1}(-\lambda(\Delta t)^\alpha) = O(\Delta t^{-\alpha})$. Because power-law decay approaches zero more slowly than exponential decay $e^{-\lambda\Delta t}$ as $\Delta t \to \infty$, smaller fractional orders correspond to slower long-range forgetting. Therefore, the adaptive kernel preserves the influence of semantically relevant historical interactions over longer time intervals, while semantically dissimilar interactions approach the exponential-forgetting regime.

B Complementary Experiments

B.1 Overall Performance Analysis for Dynamic Link Prediction

Due to space constraints, the detailed AP results, including standard deviations, are presented in Table 4. ATD-DyG achieves strong overall performance across the seven datasets, obtaining an average rank of 1.1 in both transductive and inductive settings.

Table 4: The AP metrics of the seven datasets under both transductive and inductive settings.

Datasets	Setting	JODIE	DyRep	TGAT	TGN	CAWN	EdgeBank	TCL	AdaDy-GNN	GraphMixer	DyGFormer	DyG-Mamba	ATD-DyG
UCI	Transductive	87.65±1.85	70.24±0.32	79.55±0.83	90.69±0.45	94.35±0.11	76.20±0.00	88.12±2.73	91.91±0.01	93.25±0.57	95.76±0.15	96.14±0.14	**96.15±0.11**
	Inductive	79.86±1.48	57.48±1.87	79.54±0.48	88.12±2.05	92.73±0.06	-	87.36±2.03	82.58±0.01	91.19±0.42	94.54±0.12	93.77±0.15	**95.36±0.11**
Wikipedia	Transductive	96.69±0.25	95.23±0.50	96.95±0.17	98.42±0.05	98.65±0.04	90.37±0.00	96.47±0.16	98.60±0.00	97.25±0.03	98.82±0.02	99.03±0.15	**99.37±0.11**
	Inductive	94.82±0.20	92.43±0.37	96.22±0.07	97.83±0.04	98.24±0.03	-	96.22±0.17	98.37±0.01	96.65±0.02	98.59±0.02	98.86±0.19	**99.61±0.08**
LastFM	Transductive	68.54±2.95	70.79±1.87	73.76±0.45	78.69±2.71	86.58±0.10	79.29±0.00	65.64±2.52	91.23±1.14	76.51±0.24	92.07±0.28	93.74±0.21	**95.27±0.13**
	Inductive	81.61±3.82	83.02±1.48	78.36±0.31	81.45±4.29	89.42±0.07	-	73.53±1.66	91.08±1.41	82.11±0.42	94.23±0.09	94.79±0.77	**95.71±0.07**
Enron	Transductive	79.10±0.85	82.02±3.07	72.58±0.79	85.33±1.05	89.56±0.09	83.53±0.00	79.70±0.71	79.73±0.17	82.25±0.16	92.47±0.12	**95.61±0.11**	94.20±0.11
	Inductive	80.72±1.34	74.55±3.95	67.05±1.51	77.94±1.02	86.35±0.51	-	76.14±0.79	81.91±2.71	75.88±0.48	**89.76±0.34**	88.01±2.53	88.67±0.13
USLegis.	Transductive	75.05±1.52	75.34±0.39	68.52±3.16	75.99±0.58	70.58±0.48	58.39±0.00	69.59±0.48	74.75±0.52	70.74±1.02	71.11±0.59	78.04±3.52	**88.33±0.21**
	Inductive	54.93±2.29	57.78±0.71	51.00±3.11	58.63±0.37	53.17±1.20	-	52.59±0.91	61.81±2.62	50.71±0.76	54.28±2.87	67.45±3.43	**74.28±0.59**
mooc	Transductive	80.23±2.44	81.97±0.49	85.84±0.15	89.15±1.60	80.15±0.25	57.97±0.00	82.38±0.24	87.18±0.25	82.78±0.15	87.52±0.49	88.14±0.24	**92.42±0.17**
	Inductive	79.86±1.48	57.48±1.87	79.54±0.48	88.12±2.05	92.73±0.06	-	87.36±2.03	88.64±0.62	91.19±0.42	**94.54±0.12**	94.15±0.02	94.23±0.11
UNtrade	Transductive	64.94±0.31	63.21±0.93	61.47±0.18	65.03±1.37	65.39±0.12	51.26±0.00	60.41±0.00	66.22±0.24	62.21±0.03	66.46±1.29	68.51±0.17	**75.46±0.18**
	Inductive	59.65±0.77	57.02±0.69	61.03±0.18	58.31±3.15	65.24±0.21	-	62.21±0.12	67.32±0.39	62.17±0.31	64.55±0.62	65.87±0.40	**74.35±0.15**
Avg.Rank	Transductive	9.0	8.9	9.6	5.3	6.1	9.0	10.0	5.7	7.4	3.9	2.3	**1.1**
	Inductive	8.1	9.3	9.6	7.0	4.9	-	8.2	4.6	7.3	3.1	2.6	**1.1**

B.2 Cross-Scale Reinforcement Denoising Analysis

This section provides a comprehensive visualization of the denoising decisions made by the reinforcement learning module across multiple datasets. As illustrated in Figure 5, we analyze the neighbor retention rates and selection probabilities relative to temporal distances, further validating ATD-DyG's ability to effectively identify critical structural semantics while filtering out noise across various time scales.

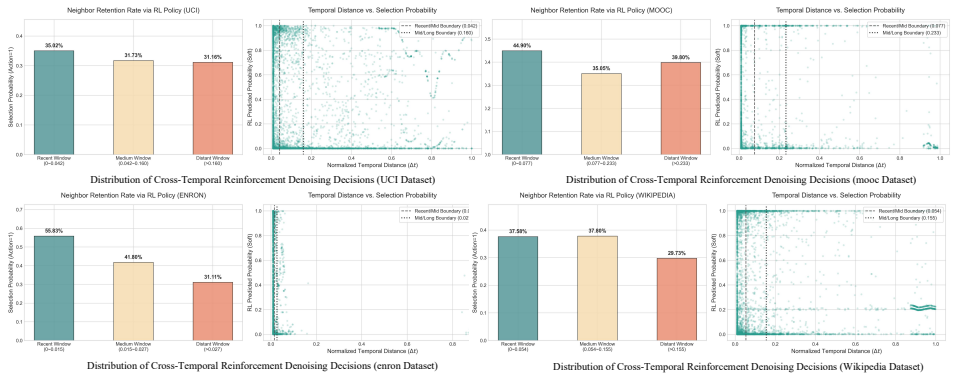
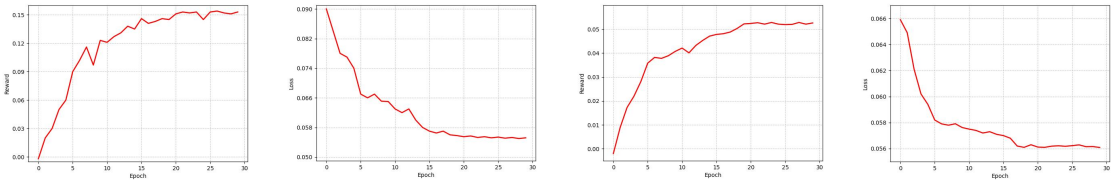


Figure 5: Cross-temporal reinforcement denoising decision distributions across multiple datasets

B.3 Training Dynamics and Stability

To evaluate the stability of our SCT-REINFORCE module, we visualize the convergence of reward and loss during training. As shown in Figure 6, the consistent upward trend in rewards and steady decline in loss across different datasets demonstrate the robust learning capability and convergence of ATD-DyG.



Reward and Loss Curves on the UCI Dataset

Reward and Loss Curves on the enron Dataset

Figure 6: Training Stability and Convergence

B.4 Impact of Candidate Evaluation Ordering

To verify whether the reinforcement learning module is sensitive to the candidate evaluation sequence, we compare our default attention-descending schedule against two alternative heuristics: random ordering and reverse attention ordering. As shown in Table 5, alternative ordering strategies cause only minor performance degradation (0.5%–0.9% AUC drop), demonstrating that our framework does not strictly depend on a specific ordering. Rather than restricting the feasible subset space, the attention-guided ordering provides a deterministic evaluation schedule and empirically yields competitive performance relative to alternative ordering strategies. We omit learned ordering and bidirectional passes because they alter the core MDP formulation by requiring nested policies or violating temporal causality.

Table 5: Performance comparison under different candidate evaluation orderings (AUC %).

Setting	Ordering Strategy	UCI	Enron	MOOC	Avg. Drop
Transductive	Attention (Ours)	95.43	93.15	91.98	—
	Random Ordering	94.82	92.48	91.31	$\downarrow 0.65$
	Reverse Attention	94.51	92.23	91.12	$\downarrow 0.89$
Inductive	Attention (Ours)	93.74	90.38	93.13	—
	Random Ordering	93.19	89.84	92.56	$\downarrow 0.55$
	Reverse Attention	92.91	89.49	92.27	$\downarrow 0.86$

B.5 Training Stability Across Random Seeds

Since reinforcement learning algorithms can be sensitive to random initialization, we further evaluate the training stability of ATD-DyG by repeating the experiments across five different random seeds (Seeds 0 to 4) on UCI, Enron, and MOOC datasets under both transductive and inductive settings. As shown in Table 6, the standard deviation of AUC scores across all benchmarks remains strictly below $\pm 0.2\%$, indicating that the policy-gradient optimization with the self-critical baseline remains stable across different random initializations.

Table 6: Performance stability across 5 random seeds (AUC %).

Dataset	Setting	Seed 0	Seed 1	Seed 2	Seed 3	Seed 4	Mean ± Std
UCI	Trans.	95.27	95.60	95.43	95.58	95.24	95.42 ± 0.17
	Induc.	93.63	93.85	93.74	93.59	93.89	93.74 ± 0.13
Enron	Trans.	93.33	93.00	93.15	93.05	93.22	93.15 ± 0.13
	Induc.	90.55	90.48	90.38	90.10	90.39	90.38 ± 0.17
MOOC	Trans.	91.80	92.15	91.98	92.08	91.89	91.98 ± 0.14
	Induc.	92.97	93.20	93.13	93.05	93.30	93.13 ± 0.13

B.6 Impact of Embedding Dimension

As illustrated in Table 7, the model performance exhibits a consistent upward trend as the dimensionality increases. Specifically, the improvement becomes marginal when the dimension reaches 128 compared to the 64-dimensional setting. To strike a balance between computational efficiency and accuracy, we ultimately adopt $d = 64$ as the final embedding dimension.

Table 7: Impact of Embedding Dimension.

Settings	AUC (%)			AP (%)		
	$d = 32$	$d = 64$	$d = 128$	$d = 32$	$d = 64$	$d = 128$
Transductive	93.92	95.43	95.85	95.01	96.15	96.78
Inductive	92.36	94.74	95.13	93.69	95.36	95.88

B.7 Hyperparameter sensitivity of the adaptive semantic evolution kernel

As illustrated in Figure 7, experiments demonstrate that the performance of the *Adaptive Semantic Evolution Kernel* is strictly constrained by a physical trade-off between "long-tail memory retention" and "transient noise filtering." While λ and σ_i respectively define the lower bound of temporal receptive field decay and the aggressiveness of semantic filtering, any deviation from the optimal distribution leads to either historical semantic fracture or structural noise overload. More importantly, the fractional truncation order K exhibits severe diminishing marginal returns: as shown in Figure 7, when $K > 20$, the marginal improvement in accuracy (AUC gain $< 0.2\%$) triggers a catastrophic expansion in memory consumption and training time. To avoid exceeding computational boundaries, this framework establishes $K = 20$ as the optimal engineering trade-off between predictive performance and system overhead.

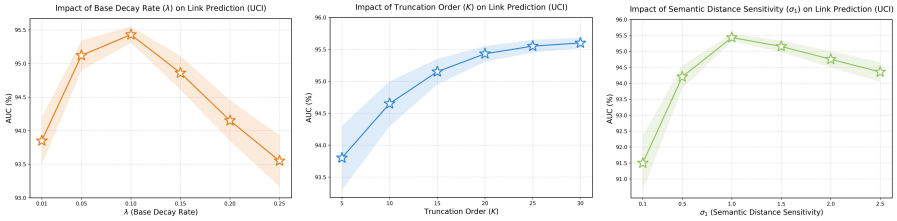


Figure 7: Training Stability and Convergence

B.8 Hardware Specification and Environment

All experiments are conducted on a professional computing server to ensure consistency and efficiency. The detailed hardware and software configurations are summarized as follows:

- **Operating System**: Ubuntu 22.04 LTS.
- **CPU**: Intel Xeon Gold 6248R @ 3.00GHz (24 cores).
- **GPU**: NVIDIA RTX A6000 with 48GB VRAM.
- **Software Environment**: Python 3.10, PyTorch 2.1.0, and CUDA 12.1.

C Datasets and Baselines

C.1 Datasets Details

- **UCI**: This dataset simulates a Facebook-like communication network among students at the University of California, Irvine. Edges denote timestamped interactions recorded with one-second temporal precision, and the dataset contains no additional features.
- **Wikipedia**: This dataset records a month of editing activity on Wikipedia pages. Editors and wiki pages are modeled as nodes, while edges represent timestamped edit requests annotated with LIWC feature vectors.
- **LastFM**: This network tracks the listening habits of 1,000 users for the top 1,000 most popular tracks over a 30-day window. Nodes correspond to listeners and songs, with edges denoting user-listens-to-song relationships without extra attributes.
- **Enron**: This dataset consists of approximately 50,000 email exchanges between Enron employees over a three-year span. Employees serve as nodes and edges designate email correspondences; no additional attributes are included.
- **USLegis.**: This dataset captures co-sponsorship behaviors among members of the US Senate. Senators are treated as nodes, and edge weights quantify the frequency of co-sponsoring a bill within a specific congressional session.
- **mooc**: This dataset characterizes a student interaction network within an online course platform. Nodes represent students or academic content units, and edges signify access events containing four specific interaction features.
- **UNtrade**: This weighted and directed network maps international trade in food and agricultural products among 181 nations over 30 years. Countries serve as nodes, and edge weights represent the normalized total value of imports or exports between nations.

C.2 Baseline Details

- **JODIE** Kumar et al. (2019): This model targets bipartite networks with instantaneous user-item interactions. It employs two coupled recurrent neural networks (RNNs) to update user and item representations and uses a projection operation to predict future trajectories.
- **DyRep** Trivedi et al. (2019): It features a custom RNN to update node representations upon observing new edges. This approach incorporates a temporal attention mechanism to assign weights to neighbors at each timestamp, effectively modeling temporal dependencies.
- **TGAT** Xu et al. (2020b): This framework aggregates features from a node's temporal-topological neighbors using a self-attention mechanism. It includes a time-encoding function to capture complex temporal patterns in dynamic network structures.

- **TGN** Rossi et al. (2020): It maintains an evolving memory for each node, which is updated through a message function, aggregator, and memory updater. An embedding module then generates temporal node representations to balance efficiency and expressiveness.
- **CAWN** Xu et al. (2020a): This method extracts multiple causal anonymous walks for each node to explore network dynamics. It employs RNNs to encode these walks and aggregates them to form the final node representation, emphasizing causality.
- **EdgeBank** Poursafaei et al. (2022): A memory-based approach designed for transductive dynamic link prediction. It stores observed interactions in a memory unit and uses simple retention strategies to predict future links, offering a lightweight solution.
- **TCL** Wang et al. (2021a): It generates node interaction sequences via a breadth-first search on temporal dependency interaction sub-graphs. A graph transformer with cross-attention is then used to model the interdependencies between interacting nodes.
- **AdaDy-GNN** Li et al. (2024): An adaptive framework that focuses on adjusting to evolving graph structures. It utilizes dynamic convolution kernels and adaptive temporal modeling to capture the non-stationary nature of interaction sequences.
- **GraphMixer** Cong et al. (2023): This model leverages a fixed-time encoding function integrated into an MLP-Mixer-based link encoder. It uses a node encoder with neighbor mean-pooling to efficiently summarize node features from temporal links.
- **DyGFormer** Yu et al. (2023): A Transformer-based architecture that focuses on first-hop interactions. It introduces a neighbor co-occurrence encoding scheme via a patching method to capture long-term correlations between source and destination sequences.
- **DyG-Mamba** Yuan et al. (2025): This state-of-the-art model utilizes selective State Space Models (SSM) to process interaction sequences. It achieves linear complexity in capturing long-range temporal dependencies, significantly improving efficiency on large-scale dynamic graphs.