Proceedings of Machine Learning Research **318** (2026)
The 39th Canadian Conference on Artificial Intelligence
Canadian AI 2026
25-29 May 2026,Vancouver,BC

Text2Edge: Language-Aware Temporal Graph Transformer for Dynamic Link Prediction

Nahid Abdolrahmanpour Holagh[†,*], Mahdis Saeedi[†], Ziad Kobti[†]
[†] School of Computer Science, University of Windsor, ON, Canada

Abstract

Dynamic link prediction in information networks (e.g., email, citation, social, and Wikipedia graphs) requires jointly modeling evolving topology and node-level semantics. However, incorporating language signals directly into temporal attention remains an open challenge. Many existing approaches either ignore textual information or attach static language features outside the attention mechanism, while naively using LLM-derived embeddings can be computationally costly and unstable without structural grounding. We introduce **Text2Edge**, a language-aware graph transformer that injects pretrained language representations into edge-sparse temporal attention, enabling semantic signals to influence how dynamic edges are weighted over time while preserving computational efficiency. Unlike purely language-based models or structure-only transformers, Text2Edge integrates semantic and structural information through a gated fusion mechanism, allowing the model to adaptively balance topology and language signals. To understand the role of semantics in dynamic link prediction, we conduct a controlled comparison between structural, semantic, and hybrid approaches. We evaluate Text2Edge alongside a strong structure-only transformer baseline (LPFormer) and language-augmented variants using BERT and LLaMA embeddings across four dynamic graph datasets. Our results show that structure-only models tend to plateau early, while semantic-aware models continue improving, indicating that semantic signals are critical in evolving real-world networks. The unified Text2Edge framework achieves the best overall performance, demonstrating that aligning pretrained language representations with edge-sparse temporal reasoning improves ranking quality and robustness without densifying the graph or fine-tuning the language encoder.

Keywords: dynamic link prediction, temporal graphs, graph transformers

1. Introduction

Large Language Models (LLMs) have recently demonstrated effective semantic modeling and generative capacity over textual information, leading to their adoption in a broad range of applications beyond traditional natural language processing. A significant portion of this textual information, however, does not exist in isolation but is embedded within relational systems where entities interact through structured connections. Much of the data generated on the Web is therefore inherently relational and naturally represented as graphs, where entities interact through links and are frequently associated with rich textual attributes and evolving interaction histories [1]. Examples include social platforms [2], citation networks [3, 4], email communication graphs [5], and community-driven knowledge platforms such as Wikipedia and Wikidata [6], where user-generated content, hyperlinks, and relationships are continuously created and updated over time. In such networks, semantic signals evolve alongside the structure. For example, in email communication networks, the topics discussed by individuals change as projects and organizational priorities shift; in citation networks, the thematic focus of research influences how future citations form. These characteristics create an opportunity to leverage LLMs not only as text encoders but also as semantic reasoning engines that can extract high-level contextual signals from network-associated text.

Predicting future interactions is a core challenge in such networks, where platforms such as social networks must determine who will connect or collaborate based on both past activity and shared interests; in citation networks, it involves anticipating which papers

[*] abdolran@uwindsor.ca

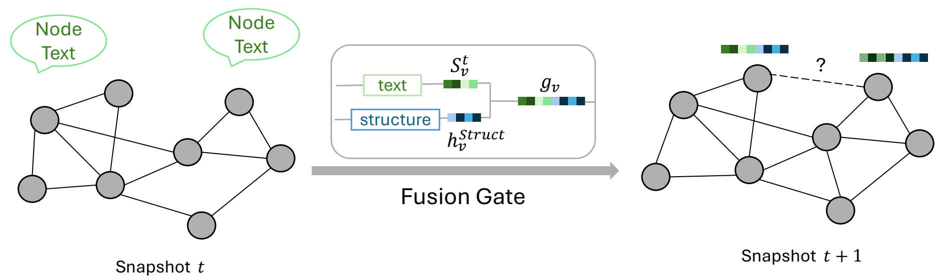


Figure 1. Overview of the proposed Text2Edge framework.

will cite one another; in email systems, which users will reply to or forward messages; and in collaborative knowledge bases like Wikipedia or Wikidata, which entities will become linked through newly created relationships [3, 7]. In these contexts, links are shaped not only by structural patterns and historical interactions but also by the underlying semantic compatibility between entities, reflected in their textual content, leading to approaches motivating integration of both topological information and contextual semantic cues to capture the full set of factors driving network evolution.

To date, prior research using LLMs for link prediction has focused mainly on knowledge graphs and heterogeneous networks [8–11], or on graphs enriched with textual attributes where prompting strategies guide how LLMs aggregate information to refine node representations [8, 12]. In these settings, semantic information is often incorporated as an auxiliary component, and the interaction between graph topology and LLM-guided semantics is not integrated within the relational reasoning process. Moreover, most existing approaches assume static graph structures, overlooking the temporal evolution that is fundamental to real-world web systems, where both interactions and associated content change continuously. Although a limited number of studies have begun exploring the integration of language models with temporal knowledge graph modeling [13, 14], Dynamic Link Prediction (DLP) with unified structural–temporal modeling and LLM-derived semantic reasoning remains largely underexplored.

At the same time, temporal graph transformers have significantly advanced DLP by modeling long-range dependencies while maintaining computational efficiency through edge-sparse attention [15], and studies show that structure-aware temporal attention and topology-aware priors enhance forward-in-time link ranking [15–17]. Unlike traditional Graph Neural Networks (GNNs), which struggle in sparse or rapidly evolving temporal graphs due to noise propagation [18], temporal graph transformers restrict attention to observed edges, enabling stable long-range reasoning. However, both approaches primarily rely on the evolving topology and remain structure-centric, limiting their ability to leverage semantic information. To address these limitations, we propose Text2Edge, a unified language-aware temporal graph transformer framework for DLP that integrates pretrained language representations directly into edge-sparse temporal attention. Unlike prior temporal graph transformer studies that focus only on topology, Text2Edge integrates LLM-derived embeddings into the attention mechanism, allowing textual similarity to modulate both message passing and edge scoring.

The main contributions of this work are as follows:

- We introduce a unified framework that integrates LLM-derived semantic embeddings with temporal graph transformer representations, enabling semantic context to directly influence relational reasoning in evolving networks.
- We design an adaptive edge-level gating mechanism that dynamically balances structural and semantic information, improving prediction accuracy in sparse and rapidly changing graphs.

- We comprehensively evaluate Text2Edge against structure-only and semantic-only baseline across four dynamic graph datasets. Using Area Under the ROC Curve (AUC) and Average Precision (AP) as evaluation metrics, which capture global ranking quality and precision–recall trade-offs, Text2Edge consistently achieves superior performance, demonstrating the benefit of jointly modeling structural, temporal, and semantic information.

2. Related Work

Dynamic link prediction addresses the problem of forecasting future interactions in graphs whose connectivity evolves over time. Early work in this area relied on heuristic proximity measures and their temporal extensions, while later approaches adopted GNNs and hybrid architectures to learn representations from node neighborhoods and historical interactions.

Despite differences in modeling choices, most DLP methods share a common assumption that says link formation is primarily explained by structural connectivity patterns and their temporal evolution. As a result, nodes are represented mainly by how they are connected rather than by their intrinsic attributes or semantic content. This structure-centric perspective underlies much of the existing DLP literature and has been shown to limit generalization, as models often capture graph-specific structural regularities that do not transfer well across networks or evolving conditions [7, 19, 20]. Even recent advances continue to refine topology-driven representations through improved temporal encodings or neighborhood modeling, rather than incorporating semantic information associated with nodes or interactions [21].

To overcome the locality and expressiveness limitations of earlier GNN-based models, transformer-based architectures have recently been introduced for link prediction. By replacing fixed neighborhood aggregation with attention mechanisms, these models enable adaptive modeling of higher-order structural dependencies beyond immediate graph proximity. A representative example is LPFormer, a structure-only graph transformer that learns link-specific structural representations by attending over local, global, and positional structural contexts [15]. In particular, LPFormer explicitly identifies a key limitation of prior link prediction methods: existing models often rely on fixed pairwise encodings with a strong inductive bias, implicitly assuming that the same underlying structural factors govern all links.

To address this limitation, LPFormer learns adaptive, link-specific representations through attention, allowing different structural factors, such as local connectivity, global context, and positional information, to be weighted differently for each candidate link [15]. By moving beyond fixed proximity heuristics and uniform encodings, LPFormer establishes a strong and principled reference point for structure-only link prediction. Subsequent work further extends topology-driven attention by incorporating auxiliary structural graphs, higher-order connectivity patterns, or topology-aware aggregation strategies. For example, AuxGT enriches structural representations by jointly modeling the original graph and an auxiliary graph capturing higher-order proximity and temporal co-occurrence patterns [16], while TopDyG emphasizes topology-aware aggregation to capture how connectivity patterns evolve over time [17].

Although these methods substantially improve the modeling of long-range structural dependencies, their attention mechanisms remain driven exclusively by connectivity patterns and positional encodings, without incorporating semantic information associated with nodes or interactions. In parallel, a growing body of research has explored incorporating semantic information into graph learning. A common strategy employs pretrained language models to encode node-associated text, which is then appended to node features prior to graph learning. This feature-level fusion has been applied in temporal knowledge graph completion and related tasks, where semantic embeddings enrich node representations but do not influence how interactions are modeled [22].

Other studies investigate the use of Large Language Models (LLMs) for reasoning over graphs through prompt-based formulations, converting graph structures into textual descriptions for language-based inference [23]. Although such approaches demonstrate that LLMs can capture certain structural and temporal patterns, they operate outside graph neural architectures and do not scale effectively to large graphs. More broadly, surveys of LLM–graph integration categorize existing methods into paradigms such as LLMs as feature generators, GNNs as prefixes to LLMs, or loosely coupled fusion frameworks [1]. Across these paradigms, semantic information is typically injected at the feature or module level, while relational reasoning remains governed by graph-based attention or message passing. Even approaches that leverage LLMs to generate symbolic rules or temporal constraints for knowledge graph reasoning maintain a separation between semantic processing and structural interaction modeling [24].

Taken together, existing research reveals a clear gap between advances in structure-centric link prediction and semantic-aware graph learning. However, prior work does not allow semantic signals to directly influence the attention-based interaction modeling that underlies DLP. In contrast, as shown in Figure 1, Text2Edge integrates LLM-derived semantic embeddings directly into attention-based interaction modeling, enabling semantic similarity and structural connectivity to jointly determine how evolving relationships are weighted and propagated.

3. Methodology

We propose Text2Edge, a language-aware temporal graph transformer for dynamic link predictio. The model is designed for evolving networks in which both interaction structure and node-associated text change over time. Rather than treating text as static side information, Text2Edge allows semantic signals to directly influence relational reasoning within the structural encoder.

We consider a temporally evolving graph represented as an ordered sequence of snapshots $\mathcal{G} = \{G_1, \ldots, G_T\}$. Each snapshot $G_t = (V_t, E_t, X_t)$ consists of active nodes V_t, observed edges $E_t \subseteq V_t \times V_t$, and node-associated textual content $X_t = \{x_v^t \mid v \in V_t\}$. A pretrained language model (e.g., BERT or LLaMA) encodes each node's text into a semantic embedding $s_v^t \in \mathbb{R}^d$, which remains frozen during training. The DLP objective is to learn a function f_θ that predicts future edges, i.e., $f_\theta(G_t) \to E_{t+1}$. The model is trained on ordered transitions $(G_1 \to E_2), (G_2 \to E_3), \ldots, (G_{T-2} \to E_{T-1})$, ensuring a strictly forward-looking setting with no access to future information during encoding.

Structural–Temporal Graph Encoding: Text2Edge models interaction dynamics using an edge-sparse graph transformer. Unlike conventional multi-hop GNNs, attention is limited to observed edges, improving stability in sparse and rapidly evolving graphs.

Node representations are initialized from semantic embeddings, $h_v^{(0)} = W_0 s_v^t$, where W_0 projects language features into the model dimension d. At transformer layer l, node u aggregates information from its observed neighbors $v \in \mathcal{N}_t(u)$ only, i.e,

$$h_u^{(l+1)} = \mathrm{AGG}\big(\{\alpha_{uv}^{(l)} W_l h_v^{(l)}\}_{v \in \mathcal{N}_t(u)}\big) \tag{3.1}$$

The attention weights $\alpha_{uv}^{(l)}$ determine how much each neighbor influences the update. Restricting attention to existing edges preserves computational efficiency and prevents noise propagation from unreliable regions of the graph.

The overall training and inference procedure of Text2Edge is summarized in Algorithm 1.

Algorithm 1 Text2Edge

Require: Temporal snapshots $\{G_t = (V_t, E_t), X_t\}_{t=1}^{T}$, frozen LLM, graph transformer $\mathcal{T}$
  Initialize: model parameters θ
  for $t = 1 \cdots T{-}2$ **do**
    Encode node text: $s_v^t \leftarrow \mathrm{LLM}(x_v^t)$
    Compute structural embeddings: $h_v^{\text{struct}} \leftarrow \mathcal{T}(E_t, \{s_v^t\})$
    Compute fusion gate: $g_v \leftarrow \sigma(W_g[h_v^{\text{struct}} \| s_v^t])$
    Fuse embeddings: $h_v^t \leftarrow g_v h_v^{\text{struct}} + (1 - g_v) W_s s_v^t$
    Compute future edge probabilities $\hat{y}_{uv}^{t+1}$
    Compute loss $\mathcal{L}$ and update θ
  end for
  Inference:
  Reuse cached s_v^{T-1}, compute h_v^{struct} and fused h_v^{T-1}
  Compute all candidate pairs probabilities $\hat{y}_{uv}$
  Rank node pairs to predict edges for the next snapshot.

Semantic Bias in Attention: Structural signals alone do not fully explain why links form, since nodes with aligned topical focus are more likely to engage in interactions. To incorporate this, Text2Edge injects semantic similarity directly into the attention computation. For edge (u, v) we compute $\mathrm{sim}_{uv} = \cos(s_u^t, s_v^t) = \frac{(s_u^t)^\top (s_v^t)}{\|s_u^t\| \|s_v^t\|}$, and the attention logits become:

$$e_{uv}^{(l)} = \frac{(W_Q h_u^{(l)})^\top (W_K h_v^{(l)})}{\sqrt{d}} + \gamma\, \mathrm{sim}_{uv}, \tag{3.2}$$

where W_Q and W_K are learnable projections and γ is a learnable scalar controlling semantic influence. After softmax normalization,

$$\alpha_{uv}^{(l)} = \frac{\exp\left(e_{uv}^{(l)}\right)}{\sum_{w \in \mathcal{N}_t(u)} \exp\left(e_{uw}^{(l)}\right)}, \tag{3.3}$$

semantic compatibility reweights the contribution of each neighbor, allowing textual signals to guide relational reasoning within the structural-temporal encoder.

Semantic–Structural Fusion: After the final transformer layer, we obtain structural embeddings h_v^{struct}. While these capture interaction patterns, text may be more reliable for sparsely connected or newly emerging nodes. To balance the two modalities, we introduce a node-wise fusion gate:

$$g_v = \sigma\big(W_g[h_v^{\text{struct}} \| s_v^t]\big)\,, \tag{3.4}$$

where σ is a sigmoid function. The final node representation is:

$$h_v = g_v\ h_v^{\text{struct}} + (1 - g_v) W_s s_v^t. \tag{3.5}$$

The gate mechanism $g_v \in [0, 1]$ adaptively balances structural and semantic contributions for each node. When the local graph neighborhood provides strong predictive signals, g_v appears 1, emphasizing the structural embedding h_v^{struct}. Conversely, for nodes with sparse or noisy connections, g_v appears 0, allowing the semantic embedding s_v^t to dominate. This continuous weighting ensures that Text2Edge flexibly leverages structure and semantics depending on the informativeness of the local context.

Link Prediction and Training: Given the fused node representations, the probability of a future edge between nodes u and v is computed using a dot-product decoder $\hat{y}_{uv} = \sigma\big(h_u^\top h_v\big)$. For each snapshot transition $G_t \to E_{t+1}$, positive samples are edges appear in E_{t+1}, while negative samples are randomly sampled from non-connected node pairs at time t, mitigating the severe class imbalance inherent in DLP. To enforce temporal causality, Text2Edge is trained under a rolling forward protocol that sequentially optimizes model parameters using only historical snapshots up to G_{T-2}. The training objective aggregates binary cross-entropy over all snapshot transitions:

$$\mathcal{L} = \sum_{t=1}^{T-2} \sum_{(u,v)} -[y_{uv}^{t+1} log\ \hat{y}_{uv}^{t+1} + (1 - y_{uv}^{t+1}) log(1 - \hat{y}_{uv}^{t+1})]. \tag{3.6}$$

During inference, Text2Edge predicts links based on a single observed snapshot G_t. The procedure follows the training pipeline without parameter updates: (1) Encode node text to obtain semantic embeddings s_v^t. (2) Compute structural embeddings h_v^{struct} using the edge-sparse graph transformer. (3) Fuse semantic and structural embeddings using the gating mechanism to obtain final representations h_v. (4) Compute link probabilities for all candidate node pairs (u, v). (5) Rank node pairs to predict edges for the next snapshot.

Edges in E_{t+1} are never observed during inference, ensuring strictly forward-looking predictions. This approach leverages temporal patterns captured in the learned parameters, enabling accurate dynamic link prediction from a single snapshot.

Evaluation is performed strictly on the unseen transition $G_{T-1} \rightarrow E_T$, with no parameter updates. This rolling protocol preserves temporal causality and prevents leakage of future edges into training. Because parameters are shared across snapshots and no recurrent states are introduced, temporal generalization emerges from sequential supervision over historical transitions. This sequential supervision ensures that the shared parameters θ learn regularities across time, capturing how interaction patterns evolve, how text predicts future edges, and how sparsity and new nodes affect link formation.

4. Experimental Setup

To systematically evaluate the effectiveness of Text2Edge and the contribution of semantic information in dynamic link Prediction, we structure our experiments around the following research questions:

RQ1: Does incorporating semantic node embeddings improve link prediction? We aim to quantify the impact of integrating textual information using large language models. By comparing temporal graph transformer variants augmented with frozen LLM-derived node embeddings against a purely structural backbone, we evaluate whether semantic representations improve link prediction beyond what is achievable using structural signals alone.

RQ2: Does injecting semantics into the attention mechanism provide benefits beyond feature-level fusion for dynamic link prediction? Text2Edge differs from embedding augmentation by directly biasing attention scores with semantic similarity and employing a semantic–structural fusion gate. This question investigates whether this interaction-aware use of text yields measurable gains over dynamic models that use embeddings only as static node features.

RQ3: How does Text2Edge perform across diverse dynamic networks? Dynamic graphs exhibit varying levels of sparsity, node turnover, and interaction patterns. We examine whether the proposed framework consistently improves link prediction across datasets with distinct domains, including citation networks (Cit-HepTh), communication networks (Enron), social interaction graphs (Reddit), and knowledge graphs (TGBL-Wiki).

RQ4: How does Text2Edge compare to existing non-temporal baselines? We evaluate Text2Edge against LPFormer, a strong structure-only model. This comparison highlights the added value of temporal modeling combined with semantic information.

4.1. **Datasets**

We evaluate Text2Edge on four text-rich dynamic graphs spanning communication, citation, social, and hyperlink domains: Enron [25], Cit-HepTh [26], Reddit [27], and TGBL-Wiki [28]. These datasets vary in scale, sparsity, temporal resolution, and textual richness, providing a diverse benchmark for semantic-aware DLP.

Following standard practice in DLP, all datasets are processed strictly in chronological order to avoid temporal leakage. Raw interaction events are sorted by timestamp and divided into fixed consecutive temporal windows, yielding a sequence of graph snapshots that the model processes sequentially. Training is performed in a strictly forward manner, where information available at time t is used to predict interactions at time $t + 1$.

Enron is an email communication network where nodes correspond to employees and edges indicate exchanged emails. Node text is derived from email subjects and message bodies. Self-loops and duplicate edges are removed.

Cit-HepTh is a citation network from the arXiv High Energy Physics Theory corpus. Nodes are papers, edges are citations, and text comes from titles and abstracts. Citation timestamps provide natural temporal ordering.

Reddit models interactions between subreddits via cross-posts. Node-level text aggregates post titles and bodies within each snapshot. Self-links are removed.

TGBL-Wiki is a large-scale, Wikipedia extracted, hyperlink network. Nodes correspond to pages, edges represent evolving hyperlinks, and node text is page content or passages.

Across all datasets, node sets evolve over time, with new nodes appearing in later snapshots. Since Text2Edge uses frozen language models for text encoding, embeddings for newly appearing nodes can be computed immediately, supporting inductive and realistic deployment in dynamic settings. Table 1 summarizes the key statistics for each dataset.

Dataset	Domain	#Nodes	#Edges	Time Span	#Snapshots	Text Source
Enron	Email	36,692	735,324	1999–2002	100	Email subject + body
Cit-HepTh	Citation	19,679	705,614	1993–2003	96	Title + abstract
Reddit	Social	2,189	516,944	2014–2016	174	Post title + body
TGBL-Wiki	Hyperlink	9,227	15.1M	2018–2020	96	Page / passage text

Table 1. Summary statistics of the temporal datasets for link prediction.

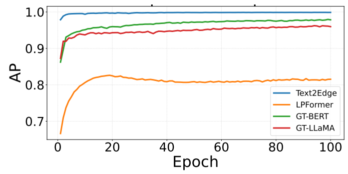
4.2. **Baselines**

To assess the benefits of incorporating semantic information, we compare Text2Edge against a set of temporal graph transformer variants and a structure-only reference, all evaluated under identical forward-in-time splits, snapshot construction, and negative sampling. This ensures that performance differences reflect the use of semantics rather than variations in preprocessing or evaluation.

- **LPFormer** [15] (Structure-only baseline) A graph transformer that performs edge-sparse attention over observed edges at each snapshot and captures temporal evolution through parameter sharing. It does not leverage node-associated text, making it a strong reference to evaluate the added value of semantic signals.
- **GT–BERT** [29, 30] (Feature-level text variant) A graph transformer-based model in which node text is encoded using a frozen BERT model and supplied as static node features to the graph transformer. Text influences only the initial node representation and does not affect the attention mechanism.
- **GT–LLaMA** [31] (Feature-level text variant) Similar to the BERT variant, but node text is embedded using a frozen LLaMA model. This allows evaluation of different language model families while keeping semantics at the feature level.

Cit-HepTh

Model	AP	AUC
GT–LLaMA	0.9623	0.9610
GT–BERT	0.9795	0.9819
LPFormer	0.8711	0.8697
Text2Edge	**0.9989**	**0.9992**



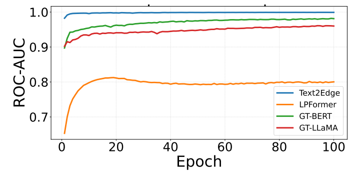
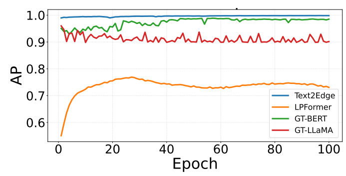


Table 2. Overall performance on Cit-HepTh (left) and training convergence behavior across epochs for all compared models (right).

Enron

Model	AP	AUC
GT–LLaMA	0.9600	0.9580
GT–BERT	0.9887	0.9890
LPFormer	0.7310	0.7832
Text2Edge	**0.9985**	**0.9991**



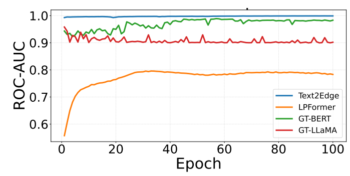


Table 3. Overall performance on Enron (left) and training convergence behavior across epochs for all compared models (right).

- **Text2Edge** (Proposed model) Extends the structure-only baseline by injecting semantic similarity into attention logits and applying a semantic–structural fusion gate to combine structural and semantic embeddings prior to link prediction. This design allows the model to adaptively leverage text when structure is sparse or noisy.

These baselines enable a systematic evaluation of how and when semantic information improves DLP, isolating the impact of attention-level integration versus feature-level usage.

4.3. Training and Evaluation Protocol

Text2Edge is trained under a strictly forward-in-time rolling evaluation protocol. Given an ordered sequence of snapshots $\{G_1, \ldots, G_T\}$, the model is optimized using only historical transitions $(G_1 \to E_2), (G_2 \to E_3), \ldots, (G_{T-2} \to E_{T-1})$, and evaluated on the next unseen transition $G_{T-1} \to E_T$. At each snapshot G_t, the model uses the observed graph structure and node text to predict edges appearing in E_{t+1}. Positive examples correspond to edges observed at time $t+1$, while negative examples are randomly sampled from non-connected node pairs at time t, addressing the extreme class imbalance typical in DLP. Loss is computed independently for each training transition using binary cross-entropy with logits, and parameters are updated sequentially across historical snapshots only, ensuring that no future edges are used during optimization. This rolling temporal regime preserves causality and enables the model to capture evolving structural and semantic regularities without introducing explicit recurrent states.

Optimization is performed using Adam, with encoder-specific learning rates. For BERT-based variants, the learning rate is 1×10^{-3}, with cosine learning-rate scheduling and global gradient clipping (maximum norm 1.0) to stabilize training. LLaMA-based variants use a lower learning rate of 2×10^{-4}, which provides stable convergence despite higher-variance embeddings; scheduling and clipping are disabled in this case. Node text is encoded once per snapshot using the frozen language model (BERT or LLaMA). The resulting embeddings are cached and optionally projected to align with the transformer's hidden dimension, reducing computational cost and maintaining consistent representation scale. Performance is reported using Average Precision (AP) and ROC–AUC. AP emphasizes early-ranking quality under extreme class imbalance, while ROC–AUC measures overall separability between future edges and non-edges. Metrics are computed exclusively on unseen future snapshots without

Reddit

Model	AP	AUC
GT–LLaMA	0.9301	0.9321
GT–BERT	0.9501	0.9523
LPFormer	0.7964	0.7409
Text2Edge	**0.9946**	**0.9935**

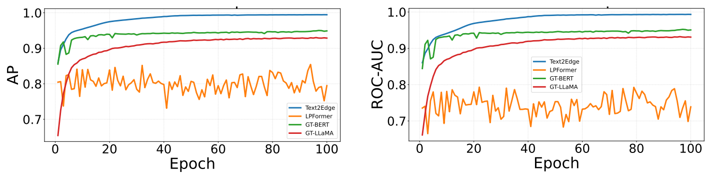


Table 4. Overall performance on Reddit (left) and training convergence behavior across epochs for all compared models (right).

TGBL-Wiki

Model	AP	AUC
GT–LLaMA	0.9898	0.9285
GT–BERT	0.9940	0.9891
LPFormer	0.8123	0.8753
Text2Edge	**0.9942**	**0.9958**

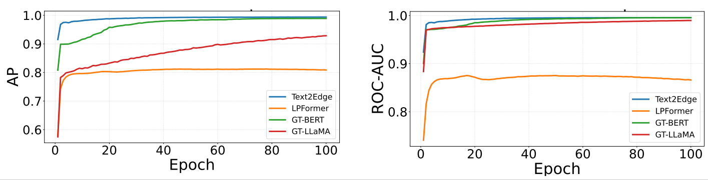


Table 5. Overall performance on TGBL-Wiki (left) and training convergence behavior across epochs for all compared models (right).

parameter updates, and all competing methods are evaluated under identical temporal splits and candidate edge sets to ensure fair and reproducible comparison.

5. Results

We evaluate Text2Edge on four dynamic graph datasets using average precision (AP) and area under the ROC curve (AUC) as evaluation metrics.

RQ1: Effect of semantics: To assess the contribution of textual information, we compare GT–BERT and GT–LLaMA against each other and against Text2Edge. Across all datasets, both LLM-augmented graph transformers outperform LPFormer, demonstrating that semantic embeddings substantially improve prediction quality. For example, from Table 2, on Cit-HepTh, GT-BERT achieves AP 0.9795 and AUC 0.9819, while GT-LLaMA achieves AP 0.9623 and AUC 0.9610, compared to LPFormer's 0.8711 AP and 0.8697 AUC. From Tables 3,4, and 5, similar trends are observed in Enron, Reddit, and TGBL-Wiki, confirming that semantic embeddings enhance temporal link prediction beyond structural information alone. Consistent improvements over the structure-only LPFormer baseline across all datasets demonstrate that semantic information provides complementary predictive signals beyond temporal graph structure. While structural models effectively capture historical interaction patterns, they are inherently limited in anticipating future links when connectivity is sparse, noisy, or rapidly evolving.

RQ2: Semantic attention vs. feature fusion: Text2Edge extends the graph transformer by introducing semantic bias in attention and a fusion gate. Compared to GT–BERT and GT–LLaMA, Text2Edge consistently achieves the highest performance across all datasets, indicating that interaction-aware use of text further improves predictions. For instance, as shown in Table 2 on Cit-HepTh, Text2Edge attains AP 0.9989 and AUC 0.9992, surpassing GT-BERT by 0.02 AP and 0.017 AUC. These results demonstrate that leveraging semantic similarity directly in attention, combined with adaptive fusion, provides measurable gains over using embeddings solely as node features. As shown in the plots of Tables 2–5, Text2Edge converges faster and maintains a stable performance margin throughout training. This indicates that injecting semantic similarity directly into the attention mechanism enables more effective interaction modeling than treating text as static node features. An additional observation concerns the relative behavior of different LLM-derived embeddings.

When incorporated at the feature level, BERT-based variants consistently outperform their LLaMA-based counterparts. This suggests that larger or more expressive language models do not automatically yield superior performance in dynamic graph settings when used naively. Instead, representation stability and alignment with structural signals appear more important than model scale alone. Text2Edge mitigates this limitation by allowing semantic information to modulate attention weights, rather than relying solely on fixed embeddings.

RQ3: Robustness across graph dynamics: The effectiveness of Text2Edge is robust across datasets with different domains and structural properties. In all four datasets, including citation networks (Cit-HepTh), email communication networks (Enron), social interactions (Reddit), and knowledge graphs (TGBL-Wiki), Text2Edge outperforms both feature-level LLM variants and the structure-only baseline, as shown in Tables 2–5. The consistent gains in AP and AUC indicate that Text2Edge generalizes well to graphs with varying sparsity, node turnover, and interaction semantics, confirming its versatility in DLP.

RQ4: Cross-dataset generalization: Text2Edge demonstrates strong and consistent performance across communication, citation, social, and web networks. Despite substantial variation in graph density, temporal dynamics, and textual characteristics, the framework remains robust, indicating its ability to capture general principles underlying the joint evolution of graph structure and node-associated text. LPFormer, a structure-only model, performs significantly worse than all temporal LLM-augmented models. Across datasets, AP and AUC differences between LPFormer and Text2Edge are substantial, for example, from Table 3, in Enron, LPFormer achieves AP 0.7310 and AUC 0.7832, whereas Text2Edge reaches 0.9985 AP and 0.9991 AUC. These results highlight the importance of temporal modeling and the added value of semantic information in predicting future links.

6. **Conclusion**

We introduced Text2Edge, a language-aware temporal graph transformer that integrates pretrained language representations directly into edge-sparse attention for dynamic link prediction. Unlike prior approaches that use textual embeddings only as static node features, Text2Edge injects semantic similarity into the attention computation itself, allowing language signals to actively modulate relational reasoning over evolving graph structure.

Through a comprehensive empirical study across four dynamic graph datasets, we show that semantics are not merely auxiliary features but significantly improve prediction quality. While feature-level LLM augmentation outperforms structure-only modeling, attention-level semantic integration consistently yields further gains, demonstrating that interaction-aware semantic bias provides robust and measurable benefits.

Our results indicate that future link formation in real-world networks depends on both temporal topology and evolving semantic compatibility. By embedding semantic reasoning directly into temporal attention, Text2Edge provides a principled bridge between large language models and dynamic graph learning.

Future work will explore parameter-efficient fine-tuning strategies, such as adapters or LoRA, to better align pretrained language models with graph-specific temporal semantics while maintaining scalability. We also plan to extend Text2Edge to heterogeneous and multi-relational dynamic graphs, including knowledge graphs with typed or weighted edges, where semantic signals may interact differently across relation types. Finally, investigating cross-graph transferability (training on one dynamic network and evaluating on another) will provide deeper insight into the robustness and generalization capacity of semantic-aware temporal transformers.

References

[1] X. Ren, J. Tang, D. Yin, N. V. Chawla, and C. Huang. "A Survey of Large Language Models for Graphs". In: *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2024*. Ed. by R. Baeza-Yates and F. Bonchi. ACM, 2024. DOI: 10.1145/3637528.3671460. URL: https://doi.org/10.1145/3637528.3671460.

[2] S. Choudhary and G. Kumar. "Enhancing link prediction in dynamic social networks through hybrid GCN-LSTM models". In: *Knowl. Inf. Syst.* 67.8 (2025), pp. 6717–6751. DOI: 10.1007/S10115-025-02430-5. URL: https://doi.org/10.1007/s10115-025-02430-5.

[3] M. Dileo, M. Zignani, and S. Gaito. "Temporal graph learning for dynamic link prediction with text in online social networks". In: (2024). DOI: 10.1007/S10994-023-06475-X. URL: https://doi.org/10.1007/s10994-023-06475-x.

[4] R. Pan, Y. Gao, and H. Wang. "A latent space model for link prediction in statistical citation network". In: *J. Multivar. Anal.* (2026). DOI: 10.1016/J.JMVA.2025.105555. URL: https://doi.org/10.1016/j.jmva.2025.105555.

[5] Q. Wang. "Link prediction and threads in email networks". In: *DSAA*.

[6] D. Vrandecic and M. Krötzsch. "Wikidata: a free collaborative knowledgebase". In: *Commun. ACM* (2014). DOI: 10.1145/2629489. URL: https://doi.org/10.1145/2629489.

[7] N. A. Holagh and Z. Kobti. "Survey of Graph Neural Network Methods for Dynamic Link Prediction". In: *The 16th International Conference on Ambient Systems, Networks and Technologies (ANT 2025) (EDI40 2025) , Patras, Greece*. Elsevier, 2025. DOI: 10.1016/J.PROCS.2025.03.057. URL: https://doi.org/10.1016/j.procs.2025.03.057.

[8] B. Bi, S. Liu, Y. Wang, L. Mei, and X. Cheng. "LPNL: Scalable Link Prediction with Large Language Models". In: Findings of ACL. Association for Computational Linguistics, 2024. DOI: 10.18653/V1/2024.FINDINGS-ACL.215. URL: https://doi.org/10.18653/v1/2024.findings-acl.215.

[9] Y. Chen and Y. Shen. "Temporal Knowledge Graph Link Prediction Using Synergized Large Language Models and Temporal Knowledge Graphs". In: Communications in Computer and Information Science. Springer, 2024. DOI: 10.1007/978-981-97-7007-6_3. URL: https://doi.org/10.1007/978-981-97-7007-6_3.

[10] Q. Lin, J. Liu, F. Xu, Y. Pan, Y. Zhu, L. Zhang, and T. Zhao. "Incorporating Context Graph with Logical Reasoning for Inductive Relation Prediction". In: ACM, 2022. DOI: 10.1145/3477495.3531996. URL: https://doi.org/10.1145/3477495.3531996.

[11] T. Le, N. Le, and B. Le. "Knowledge graph embedding by relational rotation and complex convolution for link prediction". In: (2023). DOI: 10.1016/J.ESWA.2022.119122. URL: https://doi.org/10.1016/j.eswa.2022.119122.

[12] S. Kim, S. Y. Lee, J. Yoo, and K. Shin. "'Hello, World!': Making GNNs Talk with LLMs". In: *CoRR* (2025). DOI: 10.48550/ARXIV.2505.20742. arXiv: 2505.20742. URL: https://doi.org/10.48550/arXiv.2505.20742.

[13] J. Youn and I. Tagkopoulos. "KGLM: Integrating Knowledge Graph Structure in Language Models for Link Prediction". In: 2023. DOI: 10.18653/V1/2023.STARSEM-1.20. URL: https://doi.org/10.18653/v1/2023.starsem-1.20.

[14] M. Yang, C. Yang, J. Zhu, J. Li, J. Zhang, Y. Li, and Y. Li. "SLiNT: Structure-aware Language Model with Injection and Contrastive Training for Knowledge Graph Completion". In: (2025). DOI: 10.48550/ARXIV.2509.06531. arXiv: 2509.06531. URL: https://doi.org/10.48550/arXiv.2509.06531.

[15] H. Shomer, Y. Ma, H. Mao, J. Li, B. Wu, and J. Tang. "LPFormer: An Adaptive Graph Transformer for Link Prediction". In: *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2024, Barcelona, Spain, 2024*. Ed. by R. Baeza-Yates and F. Bonchi. ACM, 2024, pp. 2686–2698. DOI: 10.1145/3637528.3672025. URL: https://doi.org/10.1145/3637528.3672025.

[16] T. Tan, X. Cao, F. Song, S. Chen, W. Du, and Y. Li. "Temporal Link Prediction via Auxiliary Graph Transformer". In: *IEEE Trans. Netw. Sci. Eng.* 11.6 (2024). DOI: 10.1109/TNSE.2024.3485093. URL: https://doi.org/10.1109/TNSE.2024.3485093.

[17] Z. et al.(2025). "Triangle Matters! TopDyG: Topology-aware Transformer for Link Prediction on Dynamic Graphs". In: *Proceedings of the ACM on Web Conference 2025, WWW 2025,*

Sydney, NSW, Australia, 28 April 2025- 2 May 2025. Ed. by G. Long, M. Blumestein, Y. Chang, L. Lewin-Eytan, Z. H. Huang, and E. Yom-Tov. ACM, 2025. DOI: 10.1145/3696410.3714564. URL: https://doi.org/10.1145/3696410.3714564.

[18] U. Alon and E. Yahav. "On the Bottleneck of Graph Neural Networks and its Practical Implications". In: *9th International Conference on Learning Representations, ICLR 2021,* OpenReview.net, 2021. URL: https://openreview.net/forum?id=i80OPhOCVH2.

[19] Z. Pan, C. Gao, F. Cai, W. Chen, X. Zhang, H. Chen, and Y. Li. "On the Cross-Graph Transferability of Dynamic Link Prediction". In: ACM, 2025. DOI: 10.1145/3696410.3714712. URL: https://doi.org/10.1145/3696410.3714712.

[20] N. A. Holagh. "Benchmarking GNN and Graph Transformer Models for Dynamic Link Prediction". In: *Social Networks Analysis and Mining*. Ed. by A. An, A. Cuzzocrea, and H. Hu. Vol. 16324. Lecture Notes in Computer Science. Cham: Springer, 2026, ??–?? DOI: 10.1007/978-3-032-14107-1_24.

[21] K. Cheng, L. Peng, P. Wang, H. Chang, J. Ye, and B. Du. "On the Scalability of Temporal Relative Positional Encoding for Dynamic Link Prediction". In: ACM, 2025, pp. 298–309. DOI: 10.1145/3711896.3737069. URL: https://doi.org/10.1145/3711896.3737069.

[22] X. Yang and J. Zhu. "Large language model with iteratively prompt for temporal knowledge graph completion". In: *Neurocomputing* 651 (2025), p. 130941. DOI: 10.1016/J.NEUCOM.2025.130941. URL: https://doi.org/10.1016/j.neucom.2025.130941.

[23] Z. et al.(2024). "LLM4DyG: Can Large Language Models Solve Spatial-Temporal Problems on Dynamic Graphs?" In: *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2024, Barcelona, Spain, August 25-29, 2024*. Ed. by R. Baeza-Yates and F. Bonchi. ACM, 2024. DOI: 10.1145/3637528.3671709. URL: https://doi.org/10.1145/3637528.3671709.

[24] J. Wang, K. Sun, L. Luo, W. Wei, Y. Hu, A. W. Liew, S. Pan, and B. Yin. "Large Language Models-guided Dynamic Adaptation for Temporal Knowledge Graph Reasoning". In: 2024. URL: http://papers.nips.cc/paper/2024/hash/0fd17409385ab9304e5019c6a6eb327a-Abstract-Conference.html.

[25] B. Klimt and Y. Yang. "Introducing the Enron Corpus". In: 2004. URL: http://www.ceas.cc/papers-2004/168.pdf.

[26] J. Leskovec, J. M. Kleinberg, and C. Faloutsos. "Graph evolution: Densification and shrinking diameters". In: *ACM Trans. Knowl. Discov. Data* 1.1 (2007), p. 2. DOI: 10.1145/1217299.1217301. URL: https://doi.org/10.1145/1217299.1217301.

[27] S. Kumar, W. L. Hamilton, J. Leskovec, and D. Jurafsky. "Community Interaction and Conflict on the Web". In: ACM, 2018, pp. 933–943. DOI: 10.1145/3178876.3186141. URL: https://doi.org/10.1145/3178876.3186141.

[28] S. Huang, F. Poursafaei, J. Danovitch, M. Fey, W. Hu, E. Rossi, J. Leskovec, M. M. Bronstein, G. Rabusseau, and R. Rabbany. "Temporal Graph Benchmark for Machine Learning on Temporal Graphs". In: 2023. URL: http://papers.nips.cc/paper_files/paper/2023/hash/066b98e63313162f6562b35962671288-Abstract-Datasets_and_Benchmarks.html.

[29] J. Devlin, M. Chang, K. Lee, and K. Toutanova. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: Association for Computational Linguistics, 2019. DOI: 10.18653/V1/N19-1423. URL: https://doi.org/10.18653/v1/n19-1423.

[30] C. Ying, T. Cai, S. Luo, S. Zheng, G. Ke, D. He, Y. Shen, and T. Liu. "Do Transformers Really Perform Badly for Graph Representation?" In: *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*. 2021. URL: https://proceedings.neurips.cc/paper/2021/hash/f1c1592588411002af340cbaedd6fc33-Abstract.html.

[31] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample. "LLaMA: Open and Efficient Foundation Language Models". In: *CoRR* abs/2302.13971 (2023). DOI: 10.48550/ARXIV.2302.13971. arXiv: 2302.13971. URL: https://doi.org/10.48550/arXiv.2302.13971.