BEYOND OBSERVED TOPOLOGY: SEMANTIC GRAPH INFERENCE AND ALIGNMENT FOR TEXT-ATTRIBUTED GRAPH LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Text-attributed graph (TAG) learning jointly models textual semantics and graph topology, typically assuming that observed edges reflect the underlying semantic structure. However, this assumption may not always hold, as connected nodes can be semantically unrelated, while semantically related nodes may lack observed connections, resulting in structural–semantic mismatch. Recent studies have explored incorporating textual semantics into graph structure, either by building upon the observed topology or by using LLMs to infer semantic relations at the edge level. While the latter offers greater flexibility, it incurs additional inference costs and limits interpretability. To address these limitations, we propose SEGIA, a SEmantic Graph Inference and Alignment approach for text-attributed graph learning. Specifically, we first introduce an uncertainty-aware semantic encoding mechanism inspired by the stochastic block model to uncover latent connectivity patterns among nodes from their textual information, yielding a semantic graph. Then, we separately encode the observed and semantic graphs to characterize their respective structural information and derive view-specific predictions. Finally, we employ optimal transport to align these predictive distributions, enabling effective information integration across the two views while enforcing prediction consistency. Extensive experiments on five public TAG benchmarks demonstrate that our proposed SEGIA outperforms state-of-the-art baselines.

1 INTRODUCTION

Text-attributed graphs (TAGs) jointly encode textual attributes and graph topology, where nodes represent entities associated with textual descriptions and edges capture their relationships. As a natural representation of interconnected textual data, TAGs have become increasingly popular and are widely adopted in real-world applications, including academic citation networks (Sen et al., 2008), web hyperlinks, and e-commerce recommendation systems (Hu et al., 2020). By jointly capturing textual semantics and relational structures, they provide a rich representation of entities and their interactions, supporting a wide range of graph learning tasks.

Existing approaches to TAG learning can be broadly categorized into two paradigms based on how textual information is incorporated. The first uses language models (LMs) as semantic encoders to transform or augment node text into semantic representations, which are subsequently fed into graph neural networks (GNNs) for neighborhood aggregation over the observed topology (Duan et al., 2023; Zhao et al., 2023). The second leverages LMs as direct predictors, constructing prompts from node text, class descriptions, and neighborhood context to infer node labels (Wang et al., 2025; Yu et al., 2025). Despite their different inference mechanisms, both paradigms exploit observed topology to establish interactions among nodes, either through graph-based message passing or neighborhood-aware language modeling.

However, the observed topology does not necessarily align with the underlying semantic relationships among nodes, resulting in a semantic–structural mismatch. Such mismatch manifests in two ways: structurally connected nodes may be semantically dissimilar, causing message passing to introduce irrelevant information (Jin et al., 2020), while semantically related nodes may lack explicit connections and thus fail to exchange useful information (Tang et al., 2023). Taking citation net-

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.

works as an example, citation edges may arise from topical relatedness, but also from methodological reuse, background discussion, or experimental comparison. Conversely, closely related papers may remain unconnected due to publication timing, citation practices, or incomplete data collection. These cases highlight the semantic–structural mismatch inherent in observed topology and suggest that meaningful connectivity may remain implicit in node textual information.

This motivates a central question: how can we uncover latent connectivity among nodes from their textual information and leverage it for graph representation learning? Rather than replacing the observed topology, we aim to preserve its structural information while uncovering additional connectivity patterns from textual semantics. This leads to a semantic graph that captures latent relationships inferred from node text and provides an additional structural view for jointly learning from observed and semantic connectivity.

Recent studies have leveraged large language models (LLMs) to enhance the structure of text-attributed graphs. For example, LLaTA (Zhang et al., 2025) frames graph structure learning as language-aware structural sampling, which organizes node texts and topological context through a hierarchical structural-entropy tree, and refines graph connectivity via tree-guided LLM reasoning and leaf-level sampling. GAugLLM (Fang et al., 2024) leverages LLMs to detect and refine potentially spurious or missing edges, constructing an augmented graph as a contrastive view for TAG learning. These studies demonstrate the potential of LLMs for discovering semantic connections beyond the observed topology. However, relying on LLMs for edge-level decisions makes the inferred connectivity sensitive to prompting and inference configurations, while the underlying decision criteria remain difficult to explicitly quantify. This limits the controllability and interpretability of the resulting semantic graph.

To address the aforementioned issues, we propose a novel **SE**mantic **G**raph **I**nference and **A**lignment approach for text-attributed graph learning, namely SEGIA. The key idea is to infer latent relational patterns from node textual information and jointly leverage them with the original graph, enabling the model to capture both observed and text-induced relationships. Specifically, we first employ a semantic variational encoder to learn latent semantic representations, posterior uncertainty, and soft cluster memberships from node textual information. We then incorporate these semantic factors into a continuous stochastic block model-inspired scoring function to estimate pairwise connectivity, with the original edges providing structural supervision, and construct a sparse semantic graph accordingly. Finally, we use two GNN branches to encode the original and semantic graphs, respectively, and introduce a symmetric optimal transport regularizer to align their predictive distributions, encouraging cross-view consistency while preserving view-specific structural information.

The main contributions are summarized as follows:

• **We identify a key limitation of relying solely on observed topology for TAG representation learning.** Observed edges may connect semantically inconsistent nodes, while latent relationships reflected in node textual information remain unmodeled.

• **We propose a stochastic semantic graph learning approach with dual-view alignment.** The approach infers a semantic graph from node text through variational semantic encoding and edge scoring inspired by a continuous stochastic block model, then jointly models it with the observed topology and aligns their predictive distributions using a symmetric optimal transport objective.

• **We evaluate the proposed SEGIA on five public TAG benchmarks.** Extensive experiments demonstrate consistent improvements over state of-the-art baselines.

2 PRELIMINARY

Text-Attributed Graphs. Formally, a TAG is represented as $\mathcal{G} = (V, \mathcal{E}, \mathbf{A}, T)$, where $V = \{v_1, v_2, \ldots, v_N\}$ is the node set, $\mathcal{E} \subseteq V \times V$ is the edge set, $\mathbf{A} \in \{0,1\}^{N\times N}$ is the corresponding adjacency matrix, and $T = \{t_i\}_{i=1}^{N}$ denotes the node textual information. In this paper, we consider label inference on TAGs, that is, given a small labeled node set V_{labeled} , the goal is to infer the labels of the remaining unlabeled nodes $V_{\text{unlabeled}} = V \setminus V_{\text{labeled}}$.

Stochastic Block Model. The Stochastic Block Model (SBM) is a latent-variable generative model in which edge formation is governed by block memberships (Holland et al., 1983; Abbe, 2017). Formally, let K denote the number of blocks, $g_i \in \{1, \ldots, K\}$ denote the block assignment of node

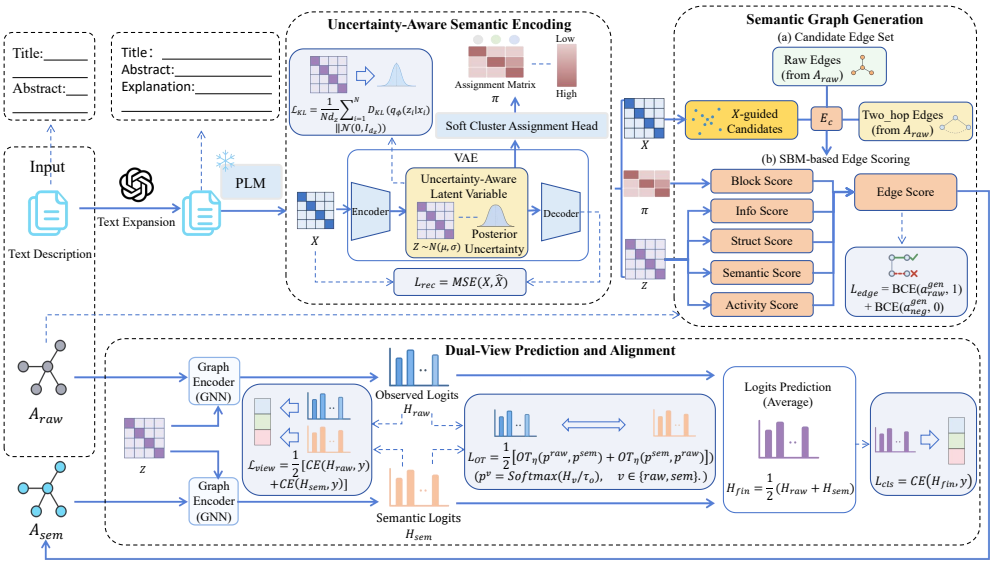

Figure 1: Architecture of our proposed SEGIA.

v_i, and $\mathbf{\Omega} \in [0,1]^{K\times K}$ specify the connection probabilities between blocks. Conditioned on the block assignments, the edge variables are mutually independent and follow the distribution as:

$$A_{i,j} \sim \text{Bernoulli}\left(\Omega_{g_i,g_j}\right), \qquad 1 \le i < j \le N. \tag{1}$$

Thus, the classical SBM assigns a common connection probability to every node pair within the same block pair. To account for variation in node degree, the degree-corrected SBM (Karrer & Newman, 2010) further introduces node-specific degree parameters $\theta_i > 0$, which modulate the connection propensity of each endpoint while preserving the block-pair connectivity pattern, written as:

$$A_{i,j} \sim \text{Bernoulli}\left(\theta_i\theta_j\Omega_{g_i,g_j}\right). \tag{2}$$

The parameters are constrained such that $0 \le \theta_i\theta_j\Omega_{g_i,g_j} \le 1$ for every node pair $i < j$. Motivated by the distinction between block-level connectivity and node-specific connectivity tendencies, we construct the semantic adjacency matrix by scoring candidate edges based on multiple factors.

3 Methodology

To address the mismatch between topology and textual semantics in TAG representation learning, we propose SEGIA, a dual-view approach that learns latent semantic relations from node textual information and jointly models them with the observed topology through symmetric optimal transport. As illustrated in Figure 1, SEGIA comprises three core components, that is, i) Uncertainty-Aware Semantic Encoding, which enriches node text with auxiliary semantic cues and learns latent semantic representations together with uncertainty estimates; ii) Semantic Graph Generation, which scores candidate edges based on multiple structural and semantic factors to construct a sparse semantic graph; as well as iii) Dual-View Prediction and Alignment, which jointly encodes the two graph views and aligns their predictive distributions.

3.1 Uncertainty-Aware Semantic Encoding

Given that node textual information can reveal latent relationships beyond the observed topology while exhibiting inherent semantic uncertainty, we introduce an uncertainty-aware semantic encoding mechanism to facilitate semantic graph generation.

LLM-Assisted Text Expansion. Raw node text often contains only a title, an abstract, or a short description, which may not fully expose the semantic cues relevant to the downstream task. To enrich the available textual information, given a node text $t_i \in T$ and a predefined set of candidate

classes $\mathcal{C} = \{c_1, \ldots, c_C\}$, we prompt an LLM to infer one or more plausible classes and provide a concise rationale highlighting the node's key topics, methods, and keywords (He et al., 2024), defined as:

$$(\widehat{Y}_i^{\mathrm{LLM}}, r_i^{\mathrm{LLM}}) = \mathrm{LLM}(t_i, \mathcal{C}, \mathrm{Prompt}), \tag{3}$$

where $\widehat{Y}_i^{\mathrm{LLM}}$ is the predicted class, and r_i^{LLM} denotes the corresponding natural-language rationale.

Then, we concatenate the raw text, predicted class, and rationale to obtain the expanded text, thereby incorporating class-related semantic cues and their supporting evidence:

$$\widetilde{t}_i = \mathrm{Concat}\Big(t_i, \widehat{Y}_i^{\mathrm{LLM}}, r_i^{\mathrm{LLM}}\Big). \tag{4}$$

Finally, we employ a frozen Pre-trained Language Model (PLM) to encode each expanded text as:

$$\mathbf{x}_i = f_{\mathrm{PLM}}(\widetilde{t}_i), \quad \mathbf{X} = [\mathbf{x}_1; \ldots; \mathbf{x}_N] \in \mathbb{R}^{N \times d_t}, \tag{5}$$

where d_t denotes the dimensionality of the node text representation.

Variational Semantic Encoding. Considering that node descriptions vary in content and specificity, the semantic information extracted from text expansion may differ in reliability and can be inevitably distorted when compressed into a finite-dimensional representation space. Treating such representations as deterministic points may obscure this inherent uncertainty and limit the reliability of subsequent semantic relation modeling. To this end, we introduce a variational autoencoder (VAE) (Kingma & Welling, 2014) to learn probabilistic latent representations. Specifically, given the initial text vector $\mathbf{x}_i$ of node v_i, we first employ VAE to transform it into a hidden representation through a linear projection and nonlinear activation as:

$$\mathbf{h}_i = \mathrm{LayerNorm}(\mathrm{Dropout}(\mathrm{ReLU}(\mathbf{W}_h \mathbf{x}_i + \mathbf{b}_h))). \tag{6}$$

Then, we adopt two independent linear projections to map it to the mean and log-variance, respectively, thereby parameterizing a diagonal Gaussian approximate posterior, defined as:

$$q_\phi(\mathbf{z}_i \mid \mathbf{x}_i) = \mathcal{N}\big(\boldsymbol{\mu}_i, \mathrm{diag}(\boldsymbol{\sigma}_i^2)\big), \tag{7}$$

where $\mathbf{z}_i, \boldsymbol{\mu}_i, \boldsymbol{\sigma}_i \in \mathbb{R}^{d_z}$, with d_z denoting the latent dimension. The corresponding posterior parameters are computed as:

$$\boldsymbol{\mu}_i = \mathbf{W}_\mu \mathbf{h}_i + \mathbf{b}_\mu, \quad \log \boldsymbol{\sigma}_i^2 = \mathrm{clip}(\mathbf{W}_\sigma \mathbf{h}_i + \mathbf{b}_\sigma, -10, 10). \tag{8}$$

It is worth noting that $\boldsymbol{\mu}_i$ and $\boldsymbol{\sigma}_i^2$ characterize the center and uncertainty of semantic distribution of node v_i in the latent space, respectively. For numerical stability, the log-variance is clipped within a predefined range. During training, we sample latent representations using the reparameterization trick, while the posterior mean is used as the deterministic semantic representation during inference. Accordingly, the latent representation used during training and inference is:

$$\mathbf{z}_i = \begin{cases} \boldsymbol{\mu}_i + \boldsymbol{\sigma}_i \odot \boldsymbol{\epsilon}, & \text{Training}, \\ \boldsymbol{\mu}_i, & \text{Inference}, \end{cases} \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{d_z}). \tag{9}$$

To prevent the latent variables from serving only the downstream classifier and losing the input semantics, we adopt a decoder g_{dec} to reconstruct the initial text vector using $\mathbf{z}_i$, namely $\widehat{\mathbf{x}}_i = g_{\mathrm{dec}}(\mathbf{z}_i)$. Mathematically, the reconstruction loss can be written as:

$$\mathcal{L}_{\mathrm{rec}} = \frac{1}{N d_t} \sum_{i=1}^{N} \|\widehat{\mathbf{x}}_i - \mathbf{x}_i\|_2^2. \tag{10}$$

Furthermore, to provide a common probabilistic reference for the approximate posterior, we use the standard Gaussian prior $p(\mathbf{z}_i) = \mathcal{N}(\mathbf{0}, \mathbf{I}_{d_z})$ and minimise the Kullback–Leibler (KL) (Blei et al., 2016) divergence between the approximate posterior and this prior:

$$\mathcal{L}_{\mathrm{KL}} = \frac{1}{2N d_z} \sum_{i=1}^{N} \sum_{r=1}^{d_z} \big(\mu_{i,r}^2 + \sigma_{i,r}^2 - \log \sigma_{i,r}^2 - 1\big). \tag{11}$$

With the reconstruction and KL divergence losses defined above, the overall VAE objective is given by their weighted combination:

$$\mathcal{L}_{\mathrm{VAE}} = \lambda_r \mathcal{L}_{\mathrm{rec}} + \lambda_{\mathrm{KL}} \mathcal{L}_{\mathrm{KL}}. \tag{12}$$

The reconstruction loss encourages the latent representation $\mathbf{z}_i$ to preserve the essential semantics of the expanded text, while the KL divergence regularizes the node-wise posterior towards the shared prior. Importantly, the resulting posterior variance $\boldsymbol{\sigma}_i^2$ provides a natural estimate of node-level semantic uncertainty, which is subsequently incorporated into semantic relation modeling.

Soft Block Memberships. To organize the latent semantic representations into a structured probabilistic space, we map $\mathbf{z}_i$ to a K-dimensional soft membership distribution $\boldsymbol{\pi}_i$ through a temperature scaled projection head. Formally, the membership distribution is computed as:

$$\boldsymbol{\pi}_i = \mathrm{softmax}\left(\frac{\mathbf{W}_c \mathbf{z}_i + \mathbf{b}_c}{\tau_c}\right), \quad \sum_{c=1}^{K} \pi_{i,c} = 1, \quad \pi_{i,c} \geq 0, \tag{13}$$

where $\mathbf{W}_c$ and $\mathbf{b}_c$ project $\mathbf{z}_i$ to the unnormalized score for the blocks, $\pi_{i,c}$ is the relative membership of node v_i in latent block c, $\tau_c > 0$ is a temperature parameter, and K denotes the number of latent semantic groups. We set $K = C$ in our implementation to match the number of classes.

3.2 Continuous SBM-Inspired Semantic Graph Construction

The observed topology provides explicit relational information but may omit semantically meaningful connections between nodes. We therefore construct a semantic graph by inferring continuous edge weights from node semantic representations. To make this process both computationally efficient and semantically informative, we first identify a sparse set of plausible candidate edges and then score them using complementary group-level, node-level, and pairwise semantic factors.

Sparse Candidate-Edge Construction. Since scoring all $N(N-1)$ ordered node pairs incurs quadratic computational and memory costs, we restrict semantic graph construction to a sparse candidate edge set, represented as:

$$\mathcal{E}_c = \mathrm{Unique}\left(\mathcal{E}_{\text{raw}} \cup \mathcal{E}_{\text{two-hop}} \cup \mathcal{E}_{\text{text-knn}}\right), \tag{14}$$

where $\mathcal{E}_{\text{raw}}$ denotes the observed edges, $\mathcal{E}_{\text{two-hop}}$ is two-hop structural links, and $\mathcal{E}_{\text{text-knn}}$ contains the k-nearest neighbors of each node selected according to cosine similarity in the textual semantic space. The $\mathrm{Unique}(\cdot)$ operation removes duplicate edges.

Node-Level Semantic Statistics. The candidate set only identifies plausible node pairs, while the connectivity of each pair is determined by their semantic characteristics. To capture these characteristics, we derive three node-level statistics from the soft block memberships and variational posterior, including membership ambiguity, latent semantic uncertainty, and node activity. Specifically, we first use the normalized entropy of the soft block membership distribution to quantify the ambiguity of node v_i's membership:

$$s_i^{\text{info}} = -\frac{1}{\log K} \sum_{c=1}^{K} \pi_{i,c} \log(\max(\pi_{i,c}, \delta)), \tag{15}$$

where $\pi_{i,c}$ denotes the membership probability of node v_i in block c, and $\delta > 0$ is a small constant for numerical stability. The normalized entropy $s_i^{\text{info}} \in [0, 1]$ is smaller for concentrated memberships and larger for diffuse assignments.

Then, we measure latent semantic uncertainty by averaging the posterior variance as:

$$s_i^{\text{struct}} = \frac{1}{d_z} \sum_{r=1}^{d_z} \exp(\log \sigma_{i,r}^2), \tag{16}$$

where larger s_i^{struct} indicates greater uncertainty and less reliable semantic evidence.

Finally, we infer a node activity parameter from the latent representation to capture node-specific tendencies to participate in edges, allowing nodes with similar semantic memberships to exhibit distinct connectivity patterns:

$$\theta_i = \mathrm{sigmoid}\left(\mathbf{w}_\theta^\top \mathrm{Dropout}(\mathbf{z}_i) + b_\theta\right) + \delta. \tag{17}$$

Continuous SBM-Inspired Candidate-Edge Scoring. Inspired by the stochastic block model (Holland et al., 1983), we combine soft block memberships with node-level semantic statistics to

assign a continuous connectivity score to each candidate edge. Formally, for each $(v_i, v_j) \in \mathcal{E}_c$, we first compute the soft block compatibility as:

$$b_{i,j} = \boldsymbol{\pi}_i^\top \mathbf{B} \boldsymbol{\pi}_j, \tag{18}$$

where $\mathbf{B} \in \mathbb{R}^{K \times K}$ is a learnable block association matrix, with $B_{c,c'}$ indicating the connectivity tendency between blocks c and c'. Since $\boldsymbol{\pi}_i$ and $\boldsymbol{\pi}_j$ are soft memberships, $b_{i,j}$ aggregates compatibility across all block pairs. It is worth noting that we initialize $\mathbf{B}$ with 1 on the diagonal and -1 elsewhere, providing an initial preference for within-block connections.

The block compatibility captures block-level connectivity but overlooks node-level variations. We therefore incorporate the node-level statistics into the edge score. First, we model membership ambiguity as:

$$q_{i,j}^{\text{info}} = \alpha s_i^{\text{info}} s_j^{\text{info}}, \tag{19}$$

where $\alpha \geq 0$ controls the contribution of the term, which becomes larger when both endpoints have more diffuse memberships.

Subsequently, node-specific connectivity tendencies are captured through an activity correction:

$$q_{i,j}^{\text{act}} = \lambda_a \log(\theta_i \theta_j + \delta), \tag{20}$$

where $\lambda_a \geq 0$ controls the contribution of endpoint activity, while the logarithmic form combines the activity of both endpoints, allowing node pairs with similar block memberships to receive different scores.

Beyond these node-level characteristics, the reliability of the latent semantic evidence also needs to be considered. To this end, we incorporate an uncertainty penalty:

$$\rho_{i,j}^{\text{struct}} = \lambda_s \left(s_i^{\text{struct}} + s_j^{\text{struct}} \right), \tag{21}$$

where $\lambda_s \geq 0$ controls the penalty strength. Higher uncertainty at either endpoint results in a larger penalty and thus a lower edge score.

Finally, pairwise semantic proximity is captured from the latent representations. Since soft block memberships provide only a group-level summary, we define a latent semantic distance penalty as:

$$\rho_{i,j}^{\text{sem}} = \gamma \frac{1}{d_z} \|\mathbf{z}_i - \mathbf{z}_j\|_2^2, \tag{22}$$

where $\gamma \geq 0$ controls the penalty strength (Hoff et al., 2002), such that larger latent distances incur stronger penalties.

By integrating the above factors, the connectivity score of each candidate edge can be defined as:

$$\ell_{i,j} = b_{i,j} + q_{i,j}^{\text{info}} + q_{i,j}^{\text{act}} - \rho_{i,j}^{\text{struct}} - \rho_{i,j}^{\text{sem}}. \tag{23}$$

We then apply the sigmoid function to obtain a continuous soft edge probability:

$$a_{i,j}^{\text{gen}} = \text{sigmoid}(\ell_{i,j}). \tag{24}$$

These soft edge probabilities probabilities are used as edge weights for the candidate pairs to construct a sparse semantic adjacency matrix, while non-candidate pairs are assigned zero weights. We symmetrize this matrix by adding its transpose and removing diagonal entries, and denote the result by $\mathbf{A}_{\text{sem}}$. We then add self-loops before applying symmetric normalization:

$$\widetilde{\mathbf{A}}_{\text{sem}} = \mathbf{D}_{\text{sem}}^{-1/2} (\mathbf{A}_{\text{sem}} + \mathbf{I}) \mathbf{D}_{\text{sem}}^{-1/2}, \tag{25}$$

where $\mathbf{D}_{\text{sem}}$ is the degree matrix of $\mathbf{A}_{\text{sem}} + \mathbf{I}$.

3.3 Dual-View Prediction and Alignment

Given the distinct perspectives captured by the observed and semantic graphs, we obtain branch-specific class logits and align their predictive distributions to encourage consistent predictions while retaining separate graph views.

Specifically, we employ two structurally identical two-layer graph convolutional networks (GCNs) (Kipf & Welling, 2017) with independent parameters for the observed and semantic graph branches.

Both branches take the latent semantic representation $\mathbf{Z}$ as their input features and produce node-level class logits $\mathbf{H}^{\mathrm{raw}}, \mathbf{H}^{\mathrm{sem}} \in \mathbb{R}^{N\times C}$:

$$\mathbf{H}^{\mathrm{raw}} = f_{\phi_r}(\widetilde{\mathbf{A}}_{\mathrm{raw}}, \mathbf{Z}), \quad \mathbf{H}^{\mathrm{sem}} = f_{\phi_s}(\widetilde{\mathbf{A}}_{\mathrm{sem}}, \mathbf{Z}), \tag{26}$$

where $\widetilde{\mathbf{A}}_{\mathrm{raw}}$ and $\widetilde{\mathbf{A}}_{\mathrm{sem}}$ denote the normalized adjacency matrices of the observed and semantic graphs, respectively, and f_{ϕ_r} and f_{ϕ_s} are the corresponding GCN classifiers.

The final node-level class logits are computed as the element-wise average of the observed-graph and semantic-graph branch outputs:

$$\mathbf{H}^{\mathrm{fin}} = \frac{1}{2}\left(\mathbf{H}^{\mathrm{raw}} + \mathbf{H}^{\mathrm{sem}}\right). \tag{27}$$

Cross-View Predictive Consistency. Since the observed and semantic graphs capture different relational information and may produce different predictions for the same node, we align their predictive distributions to encourage cross-view prediction consistency. Formally, we apply temperature scaling to the branch-specific logits to obtain class-probability distributions:

$$\mathbf{P}^{\mathrm{raw}} = \mathrm{softmax}(\mathbf{H}^{\mathrm{raw}}/\tau_o), \quad \mathbf{P}^{\mathrm{sem}} = \mathrm{softmax}(\mathbf{H}^{\mathrm{sem}}/\tau_o), \tag{28}$$

where $\tau_o > 0$ controls the concentration of the predictive distributions.

To align these distributions, we use a Sinkhorn-based transport cost (Cuturi, 2013; Courty et al., 2017). Let $\mathbf{M} \in \mathbb{R}^{C\times C}$ be the class-wise cost matrix, with zero cost for matching the same class and unit cost for different classes. Let $\mathbf{p}_i^{\mathrm{raw}}$ and $\mathbf{p}_i^{\mathrm{sem}}$ denote the class-probability distributions of node v_i from the observed and semantic graph views, respectively. For distributions $\mathbf{p}$ and $\mathbf{q}$, let $\mathcal{U}(\mathbf{p}, \mathbf{q})$ be the set of nonnegative transport plans with row and column marginals $\mathbf{p}$ and $\mathbf{q}$. We define the entropy-regularized transport plan as

$$\mathbf{\Pi}_\eta^\star(\mathbf{p}, \mathbf{q}) = \underset{\mathbf{\Pi}\in\mathcal{U}(\mathbf{p},\mathbf{q})}{\arg\min} \left[\langle \mathbf{\Pi}, \mathbf{M}\rangle_F + \eta \sum_{a=1}^{C}\sum_{b=1}^{C} \Pi_{a,b}(\log \Pi_{a,b} - 1)\right]. \tag{29}$$

We use the approximate transport plan to define the corresponding transport cost as

$$\mathrm{OT}_\eta(\mathbf{p}, \mathbf{q}) = \left\langle \widehat{\mathbf{\Pi}}_\eta^{(T_{\mathrm{SK}})}(\mathbf{p}, \mathbf{q}), \mathbf{M}\right\rangle_F, \tag{30}$$

where $\eta > 0$ controls the entropy regularization and $\widehat{\mathbf{\Pi}}_\eta^{(T_{\mathrm{SK}})}$ approximates $\mathbf{\Pi}_\eta^\star$ after T_{SK} Sinkhorn iterations. The entropy term affects the transport plan, while the alignment loss uses only the resulting transport cost. We average the bidirectional transport costs over all nodes to obtain

$$\mathcal{L}_{\mathrm{OT}} = \frac{1}{2N}\sum_{v_i\in V}\left[\mathrm{OT}_\eta(\mathbf{p}_i^{\mathrm{raw}}, \mathbf{p}_i^{\mathrm{sem}}) + \mathrm{OT}_\eta(\mathbf{p}_i^{\mathrm{sem}}, \mathbf{p}_i^{\mathrm{raw}})\right]. \tag{31}$$

Training Objective. The final and branch-specific class logits are jointly supervised on the training nodes using cross-entropy (CE) loss:

$$\mathcal{L}_{\mathrm{cls}} = \mathrm{CE}(\mathbf{H}^{\mathrm{fin}}, \mathbf{Y}), \tag{32}$$

$$\mathcal{L}_{\mathrm{view}} = \frac{1}{2}\left[\mathrm{CE}(\mathbf{H}^{\mathrm{raw}}, \mathbf{Y}) + \mathrm{CE}(\mathbf{H}^{\mathrm{sem}}, \mathbf{Y})\right]. \tag{33}$$

The semantic encoder is trained with the Gaussian KL divergence and text reconstruction losses, while the edge scorer is trained with binary cross-entropy (BCE) on observed edges and randomly sampled negative edges:

$$\mathcal{L}_{\mathrm{edge}} = \mathrm{BCE}(\mathbf{a}_{\mathrm{raw}}^{\mathrm{gen}}, \mathbf{1}) + \mathrm{BCE}(\mathbf{a}_{\mathrm{neg}}^{\mathrm{gen}}, \mathbf{0}), \tag{34}$$

where $\mathbf{a}_{\mathrm{raw}}^{\mathrm{gen}}$ and $\mathbf{a}_{\mathrm{neg}}^{\mathrm{gen}}$ denote the scores for observed edges and sampled non-edges, respectively. Combining these objectives with the cross-view OT alignment loss, the overall training objective is:

$$\mathcal{L} = \mathcal{L}_{\mathrm{cls}} + \lambda_v\mathcal{L}_{\mathrm{view}} + \lambda_{\mathrm{KL}}\mathcal{L}_{\mathrm{KL}} + \lambda_r\mathcal{L}_{\mathrm{rec}} + \lambda_e\mathcal{L}_{\mathrm{edge}} + \lambda_o\mathcal{L}_{\mathrm{OT}}, \tag{35}$$

where λ_v, λ_{KL}, λ_r, λ_e, and λ_o are trade-off coefficients balancing the corresponding loss terms.

Table 1: Node classification results in terms of accuracy (%). The best and second-best results are marked in bold and underline, respectively.

Models	Cora	Citeseer	PubMed	ArXiv-2023	OGBN-ArXiv
MLP	84.13 ± 0.31	69.99 ± 0.31	39.41 ± 0.90	70.20 ± 0.33	66.45 ± 0.33
GCN	84.97 ± 0.34	71.97 ± 0.33	50.11 ± 2.04	$\underline{74.38 \pm 0.33}$	70.76 ± 0.16
GAT	83.01 ± 0.46	72.74 ± 0.42	53.50 ± 1.15	70.46 ± 0.20	68.63 ± 0.11
GraphSage	85.56 ± 1.54	57.03 ± 1.26	82.35 ± 0.60	37.58 ± 4.52	32.22 ± 5.49
GCNII	88.51 ± 0.52	77.14 ± 1.21	89.54 ± 0.35	55.93 ± 2.52	58.36 ± 2.73
G2P2	83.26 ± 2.39	73.07 ± 1.02	81.29 ± 0.11	52.33 ± 0.26	63.89 ± 0.01
SimTEG	78.41 ± 0.40	66.19 ± 0.29	74.17 ± 0.12	63.31 ± 0.14	63.10 ± 0.40
CGT	$\underline{89.26 \pm 0.45}$	68.02 ± 0.70	78.72 ± 0.55	55.82 ± 0.85	57.40 ± 0.77
GLEM	87.11 ± 1.14	74.53 ± 1.66	85.24 ± 2.70	60.43 ± 3.95	63.35 ± 0.21
TANS	88.80 ± 0.43	77.30 ± 0.91	88.99 ± 0.30	67.78 ± 0.38	69.57 ± 0.34
LLM4NG	87.55 ± 1.17	75.80 ± 1.34	88.61 ± 0.22	70.78 ± 0.35	$\underline{73.67 \pm 0.26}$
DeloopSGNN	88.38 ± 0.86	76.70 ± 2.03	$\underline{90.13 \pm 0.17}$	59.58 ± 0.46	61.01 ± 0.31
AD-GCN	86.90 ± 1.54	$\underline{79.65 \pm 1.18}$	88.99 ± 0.16	65.87 ± 0.59	63.93 ± 0.89
AdaSpec	87.97 ± 1.40	76.72 ± 1.50	86.99 ± 0.47	58.50 ± 0.25	67.20 ± 0.26
RADE	87.75 ± 0.89	75.50 ± 1.72	89.43 ± 0.41	64.68 ± 0.43	68.44 ± 0.31
SLGNN	87.16 ± 0.88	73.45 ± 1.72	88.33 ± 0.52	64.96 ± 1.01	61.11 ± 0.24
SEGIA	$\mathbf{91.00 \pm 0.71}$	$\mathbf{81.32 \pm 1.03}$	$\mathbf{90.45 \pm 0.22}$	$\mathbf{78.20 \pm 0.26}$	$\mathbf{76.54 \pm 0.06}$

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Datasets. We evaluate SEGIA on five citation network datasets, namely Cora (Sen et al., 2008), CiteSeer (Giles et al., 1998), PubMed (Sen et al., 2008), ArXiv-2023 (He et al., 2024), and OGBN-ArXiv Hu et al. (2020). Detailed statistics of these datasets are provided in Appendix F.

Baselines. We compare our proposed SEGIA against representative baselines from two categories, including **1) Traditional Methods:** MLP, GCN (Kipf & Welling, 2017), GAT (Velickovic et al., 2018), GraphSAGE (Hamilton et al., 2017), and GCNII (Chen et al., 2020); as well as **2) Recent Works Designed for TAGs:** G2P2 (Wen & Fang, 2023), SimTEG (Duan et al., 2023), CGT (Vakil & Amiri, 2024), GLEM (Zhao et al., 2023), TANS (Wang et al., 2025), LLM4NG (Yu et al., 2025), DeloopSGNN (Li et al., 2026), AD-GCN (Hevapathige et al., 2026), AdaSpec (Liu & Wang, 2026), RADE (Saber & Salehi-Abari, 2026), and SLGNN (Zhang et al., 2026).

4.2 PERFORMANCE COMPARISON

We compare SEGIA with state-of-the-art baselines and report mean accuracy and standard deviation over five independent runs.

As shown in Table 1, SEGIA achieves the best performance on all five datasets, exceeding the strongest baseline by 1.74%, 1.67%, 0.32%, 3.82%, and 2.87%, respectively. These gains demonstrate the value of recovering text-induced relations and aligning them with the observed topology. Compared with recent TAG methods that enrich node features or propagation while remaining tied to the recorded graph, SEGIA performs particularly well on ArXiv-2023 and OGBN-ArXiv, indicating that text-derived propagation signals can complement fixed-edge propagation on these benchmarks. Compared with GCN, SEGIA improves performance by 6.03%, 9.35%, 40.34%, 3.82%, and 5.78% across the five datasets. These gaps show that the semantic graph complements observed-edge propagation with text-derived signals. Although the strongest overall baseline on ArXiv-2023 is a conventional GNN, SEGIA still exceeds the strongest TAG-specific method by 7.42%. Overall, these results support the dual-view design: the observed graph preserves reliable structural evidence, while the semantic graph supplies complementary relations inferred from text.

4.3 ABLATION STUDY

To effectively assess the contribution of each component in SEGIA, we report the mean accuracy of the following six variants over 5 runs: 1/2) w/o Semantic Graph/Observed Topology, using only

Table 2: Comparisons of our SEGIA with its six variants in terms of accuracy (%).

Variants	Cora	CiteSeer	PubMed	ArXiv-2023	OGBN-ArXiv
w/o Semantic Graph	88.19	72.10	87.88	75.07	75.92
w/o Observed Topology	79.15	78.21	89.02	76.98	73.84
w/o OT	88.01	77.89	89.32	77.57	76.40
w/o Uncertainty	90.40	78.68	89.60	77.57	76.38
w/o SBM Scoring	88.56	75.39	88.64	77.91	76.36
w/o LLM Expansion	88.56	78.84	89.45	77.41	76.29
SEGIA	**91.00**	**81.32**	**90.45**	**78.20**	**76.54**

the observed topology/semantic graph, respectively; 3) w/o OT, removing cross-view optimal transport alignment; 4) w/o Uncertainty, replacing stochastic posterior sampling with deterministic mean representations; 5) w/o SBM Scoring, replacing the SBM-based scores with normalized cosine similarity; and 6) w/o LLM Expansion, directly encoding the raw text without LLM-based expansion.

As shown in Table 2, SEGIA outperforms all variants across the five datasets. On Cora, removing the observed view causes the largest drop (11.85%), followed by removing OT (2.99%) and the semantic view (2.81%), indicating that reliable observed structure strengthens the semantic graph through alignment. On CiteSeer, the sparsest graph, removing the semantic view causes a larger loss than removing the observed topology (9.22% versus 3.11%), showing that view importance depends on the recorded topology. Additionally, Replacing SBM scoring with cosine similarity reduces PubMed accuracy by 1.81%, confirming the value of block compatibility and node activity beyond pairwise textual similarity. Moreover, On ArXiv-2023, removing the semantic view and LLM expansion reduces accuracy by 3.13% and 0.79%, respectively, supporting the contribution of category-level textual evidence. Removing uncertainty also degrades performance across all datasets, consistent with its role in down-weighting candidates supported by diffuse latent representations.

Candidate-set Ablation Study. Figure 2 compares the observed edges with candidate sets augmented by text neighbors, two-hop links, or both. As shown, incorporating text neighbors consistently improves performance across all five datasets, whereas two-hop links improve performance on CiteSeer and OGBN-ArXiv but lead to performance degradation on Cora, PubMed, and ArXiv-2023. Combining both sources achieves the best performance on every dataset, outperforming the stronger single-source configuration by 0.30% to 2.01%. These results suggest that semantic proximity and structurally inferred relations provide complementary evidence for semantic graph construction.

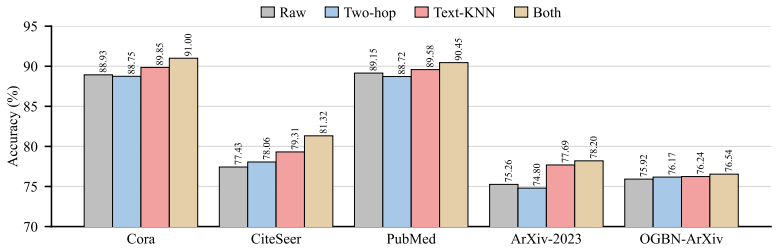

Figure 2: Accuracy (%) of the four candidate-set configurations on the five benchmarks.

5 Conclusion

In this paper, we propose SEGIA, a semantic graph inference and alignment approach for text-attributed graph learning under structural–semantic mismatch. SEGIA learns latent semantics, posterior uncertainty, and soft block memberships from enriched text using an uncertainty-aware variational encoder. It constructs a sparse semantic graph with a continuous SBM-inspired scoring mechanism that integrates block compatibility, node activity, semantic proximity, and uncertainty. Finally, SEGIA exploits the observed and inferred semantic views by aligning them through symmetric optimal transport. Experiments on five TAG benchmarks show consistent improvements over baselines, demonstrating complementary roles of observed topology and text-inferred connectivity.

AI USE STATEMENT

In this work, we used generative AI tools to assist with language polishing, improving the clarity and presentation of the manuscript, and providing limited programming support during implementation and debugging. We did not use generative AI tools to generate or fabricate experimental results, to analyze data without author verification, or to produce the core scientific contributions, including the proposed methodology, theoretical claims, and experimental conclusions. Other categories requiring disclosure are not applicable to this work. We reviewed and verified all AI-assisted text and code. In particular, all technical descriptions, mathematical formulations, experimental settings, results, and conclusions were checked by the authors; any AI-assisted code was inspected, integrated, and tested by the authors. We take full responsibility for the final content of this work, including all text, claims, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We support reproducibility by providing a detailed description of SEGIA, including its uncertainty-aware semantic encoding, continuous SBM-inspired semantic graph construction, and dual-view prediction and alignment mechanism, in Section 3. The experimental protocol, benchmark construction, baseline comparisons, and evaluation metrics are described in Section 4. We use the publicly available Cora, CiteSeer, PubMed, ArXiv-2023, and OGBN-ArXiv benchmarks, following a unified random 6:2:2 data-splitting protocol. The appendices provide dataset statistics, text-expansion prompts, and further implementation details. Upon publication, we will release the complete source code, preprocessing and text-expansion scripts, dataset splits, configuration files, and instructions for reproducing all reported experiments.

REFERENCES

Emmanuel Abbe. Community detection and stochastic block models: Recent developments. *Journal of Machine Learning Research*, 18:177:1–177:86, 2017.

Edoardo M. Airoldi, David M. Blei, Stephen E. Fienberg, and Eric P. Xing. Mixed membership stochastic blockmodels. In *Proceedings of Advances in Neural Information Processing Systems*, pp. 33–40, 2008.

Ramnath Balasubramanyan and William W. Cohen. Block-lda: Jointly modeling entity-annotated text and entity-entity links. In *Proceedings of the Eleventh SIAM International Conference on Data Mining*, pp. 450–461, 2011.

Norbert Binkiewicz, Joshua T. Vogelstein, and Karl Rohe. Covariate-assisted spectral clustering. *arXiv preprint arXiv:1411.2158*, 2014.

David M. Blei, Alp Kucukelbir, and Jon D. McAuliffe. Variational inference: A review for statisticians. *arXiv preprint arXiv:1601.00670*, 2016.

Charles Bouveyron, Pierre Latouche, and Rawya Zreik. The stochastic topic block model for the clustering of vertices in networks with textual edges. *Statistics and Computing*, 28(1):11–31, 2018.

Ming Chen, Zhewei Wei, Zengfeng Huang, Bolin Ding, and Yaliang Li. Simple and deep graph convolutional networks. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 1725–1735, 2020.

Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(9):1853–1865, 2017.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Proceedings of Advances in Neural Information Processing Systems*, pp. 2292–2300, 2013.

Yash Deshpande, Subhabrata Sen, Andrea Montanari, and Elchanan Mossel. Contextual stochastic block models. In *Proceedings of Advances in Neural Information Processing Systems*, pp. 8590–8602, 2018.

Keyu Duan, Qian Liu, Tat-Seng Chua, Shuicheng Yan, Wei Tsang Ooi, Qizhe Xie, and Junxian He. Simteg: A frustratingly simple approach improves textual graph learning. *arXiv preprint arXiv:2308.02565*, 2023.

Odilon Duranthon and Lenka Zdeborová. Neural-prior stochastic block model. *Machine Learning: Science and Technology*, 4(3):35017, 2023.

Yi Fang, Dongzhe Fan, Daochen Zha, and Qiaoyu Tan. Gaugllm: Improving graph contrastive learning for text-attributed graphs with large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 747–758, 2024.

C. Lee Giles, Kurt D. Bollacker, and Steve Lawrence. Citeseer: An automatic citation indexing system. In *Proceedings of the 3rd ACM International Conference on Digital Libraries*, pp. 89–98, 1998.

William L. Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Proceedings of Advances in Neural Information Processing Systems*, pp. 1024–1034, 2017.

Xiaoxin He, Xavier Bresson, Thomas Laurent, Adam Perold, Yann LeCun, and Bryan Hooi. Harnessing explanations: Llm-to-lm interpreter for enhanced text-attributed graph representation learning. In *Proceedings of the 12th International Conference on Learning Representations*, 2024.

Asela Hevapathige, Asiri Wijesinghe, and Ahad N. Zehmakan. Beyond fixed depth: Adaptive graph neural networks for node classification under varying homophily. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 21717–21725, 2026.

Peter D. Hoff, Adrian E. Raftery, and Mark S. Handcock. Latent space approaches to social network analysis. *Journal of the American Statistical Association*, 97(460):1090–1098, 2002.

Paul W. Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. Stochastic blockmodels: First steps. *Social Networks*, 5(2):109–137, 1983.

Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: Datasets for machine learning on graphs. In *Proceedings of Advances in Neural Information Processing Systems*, 2020.

Wei Jin, Yao Ma, Xiaorui Liu, Xianfeng Tang, Suhang Wang, and Jiliang Tang. Graph structure learning for robust graph neural networks. In *Proceedings of the 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 66–74, 2020.

Brian Karrer and M. E. J. Newman. Stochastic blockmodels and community structure in networks. *arXiv preprint arXiv:1008.3926*, 2010.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *Proceedings of the 2nd International Conference on Learning Representations*, 2014.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *Proceedings of the 5th International Conference on Learning Representations*, 2017.

Duanyu Li, Huijun Wu, Min Xie, Kai Lu, Wenzhe Zhang, Zhenwei Wu, Yong Dong, and Ruibo Wang. Deloopsgnn: Revisiting spectral gnns through the lens of spatial aggregation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 22878–22886, 2026.

Fangbing Liu and Qing Wang. Adaspec: Adaptive spectrum for enhanced node distinguishability. In *Proceedings of the Fourteenth International Conference on Learning Representations*, 2026.

Nikhil Mehta, Lawrence Carin, and Piyush Rai. Stochastic blockmodels meet graph neural networks. In *Proceedings of the 36th International Conference on Machine Learning*, pp. 4466–4474, 2019.

M. E. J. Newman and Aaron Clauset. Structure and inference in annotated networks. *arXiv preprint arXiv:1507.04001*, 2015.

Danial Saber and Amirali Salehi-Abari. Rade: Random add-drop edge as a regularizer. *arXiv preprint arXiv:2606.00757*, 2026.

Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Gallagher, and Tina Eliassi-Rad. Collective classification in network data. *AI Magazine*, 29(3):93–106, 2008.

Jianheng Tang, Weiqi Zhang, Jiajin Li, Kangfei Zhao, Fugee Tsung, and Jia Li. Robust attributed graph alignment via joint structure learning and optimal transport. In *Proceedings of the 39th IEEE International Conference on Data Engineering*, pp. 1638–1651, 2023.

Anton Tsitsulin, John Palowitch, Bryan Perozzi, and Emmanuel Müller. Graph clustering with graph neural networks. *Journal of Machine Learning Research*, 24:127:1–127:21, 2023.

Nidhi Vakil and Hadi Amiri. Controlled transformation of text-attributed graphs. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 15735–15748, 2024.

Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *Proceedings of the 6th International Conference on Learning Representations*, 2018.

Yuxiang Wang, Xiao Yan, Shiyu Jin, Quanqing Xu, Chuang Hu, Yuanyuan Zhu, Bo Du, Jia Wu, and Jiawei Jiang. Exploiting text semantics for few and zero shot node classification on text-attributed graph. In *Proceedings of the 34th International Joint Conference on Artificial Intelligence*, pp. 3462–3470, 2025.

Zhihao Wen and Yuan Fang. Augmenting low-resource text classification with graph-grounded pre-training and prompting. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 506–516, 2023.

Jianxiang Yu, Yuxiang Ren, Chenghua Gong, Jiaqi Tan, Xiang Li, and Xuecang Zhang. Leveraging large language models for node generation in few-shot learning on text-attributed graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 13087–13095, 2025.

Kaicheng Zhang, David N. Reynolds, Piero Deidda, and Francesco Tudisco. Stuart-landau oscillatory graph neural network. In *Proceedings of the ACM on Web Conference 2026*, pp. 1469–1480, 2026.

Zhihan Zhang, Xunkai Li, Lei Zhu, Guang Zeng, Bowen Fan, Yanzhe Wen, Hongchao Qin, Rong-Hua Li, and Guoren Wang. Unlocking graph structure learning with tree-guided large language models. *arXiv preprint arXiv:2503.21223*, 2025.

Jianan Zhao, Meng Qu, Chaozhuo Li, Hao Yan, Qian Liu, Rui Li, Xing Xie, and Jian Tang. Learning on large-scale text-attributed graphs via variational inference. In *Proceedings of the 11th International Conference on Learning Representations*, 2023.

A Related Work

We review two closely related directions: representation learning and semantic structure inference on text-attributed graphs, and stochastic block models with node or edge attributes.

A.1 Representation Learning on Text-Attributed Graphs

GCN (Kipf & Welling, 2017), GAT (Velickovic et al., 2018), and GraphSAGE (Hamilton et al., 2017) learn node representations through convolutional, attention-based, or sampled-neighbourhood aggregation over a supplied topology. On text-attributed graphs, they commonly treat encoded node text as input features and propagate it over observed edges. Representation quality therefore depends on both textual encoding and the recorded neighbourhoods.

Language-model integration strengthens the textual component through two-stage or joint training. For example, SimTEG (Duan et al., 2023) adapts a language model and then trains a GNN on its embeddings, whereas GLEM (Zhao et al., 2023) alternates language-model and GNN optimisation through variational expectation maximisation. TAPE (He et al., 2024) converts LLM-generated predictions and explanations into GNN features, while LLM4NG (Yu et al., 2025) uses LLMs to generate nodes for few-shot TAG learning. Despite these different integration strategies, the observed graph generally continues to determine which nodes exchange messages.

This fixed propagation structure exposes a structural–semantic mismatch. Recorded edges may connect semantically dissimilar nodes, whereas related but unconnected nodes fail to directly exchange useful information (Jin et al., 2020; Tang et al., 2023). Graph structure learning therefore extends the focus from node features to connectivity. For instance, GAugLLM (Fang et al., 2024) constructs an augmented contrastive view from structural and textual evidence, while LLaTA (Zhang et al., 2025) uses a structural-entropy tree for LLM-guided structural sampling. However, repeatedly invoking an LLM for edge-level inference can incur substantial computational cost. The remaining problem is how to infer pair-specific semantic connections from explicit, controllable evidence while preserving reliable observed relations and limiting both inference cost and noise from irrelevant edges or ambiguous text.

A.2 Stochastic Block Models for Attributed Graphs

Stochastic block models describe network connectivity through latent groups and block-pair interaction probabilities (Holland et al., 1983). Degree correction introduces node-specific connection propensities, allowing nodes in the same group to have different expected degrees (Karrer & Newman, 2010). Mixed-membership models relax the single-group assumption by assigning each node a distribution over groups and allowing its role to vary across interactions (Airoldi et al., 2008). These extensions separate group-level regularities from node-level heterogeneity and provide interpretable descriptions of overlapping or unevenly connected communities.

Attributed extensions introduce evidence beyond the adjacency matrix. Covariate-assisted spectral clustering and contextual block models combine node covariates with connectivity (Binkiewicz et al., 2014; Deshpande et al., 2018). Annotated-network models relate metadata to communities without assuming that it determines the partition (Newman & Clauset, 2015). Block-LDA and the stochastic topic block model jointly model network relations and text (Balasubramanyan & Cohen, 2011; Bouveyron et al., 2018). Thus, attributes can inform latent organisation while connectivity remains structural evidence.

Neural formulations couple learned representations with community structure through variational block models, learned structural priors, or differentiable clustering objectives (Mehta et al., 2019; Duranthon & Zdeborová, 2023; Tsitsulin et al., 2023). Their principal objectives remain explaining observed links or recovering communities. Block-pair parameters capture group-level connection tendencies, and degree correction captures node-level activity. Even when attributes inform memberships, these quantities do not usually express pair-specific semantic support between free-form node texts. The unresolved problem is how to turn block-structured regularities into controllable candidate-edge scores and use the inferred relations for node-classification propagation without discarding the observed graph.

B Theoretical Background on Variational Semantic Encoding

The semantic encoder uses an encoder–decoder architecture to obtain a stochastic representation of each text feature vector. The encoder maps the input to a distribution in latent space, and the decoder maps a sampled latent representation back to the text-feature space. This section concerns the reconstruction and KL components of this auxiliary semantic-encoding objective. The probabilistic model is described first, followed by a derivation relating the dimension-normalised training losses to a variational objective.

B.1 Latent Representation and Reconstruction Model

For a frozen text embedding $\mathbf{x}_i \in \mathbb{R}^{d_t}$, the encoder outputs a mean $\boldsymbol{\mu}_i$ and diagonal variance $\boldsymbol{\sigma}_i^2$. The approximate posterior, latent prior, and training-time reparameterisation are:

$$q_i = q_\phi(\mathbf{z}_i \mid \mathbf{x}_i) = \mathcal{N}\big(\boldsymbol{\mu}_i, \mathrm{diag}(\boldsymbol{\sigma}_i^2)\big), \qquad p(\mathbf{z}_i) = \mathcal{N}(\mathbf{0}, \mathbf{I}_{d_z}), \tag{36}$$

$$\mathbf{z}_i = \boldsymbol{\mu}_i + \boldsymbol{\sigma}_i \odot \boldsymbol{\epsilon}_i, \qquad \boldsymbol{\epsilon}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{d_z}). \tag{37}$$

The reparameterisation in Eq. (37) permits gradients to pass through sampling (Kingma & Welling, 2014). At inference time, $\mathbf{z}_i$ is replaced by the posterior mean $\boldsymbol{\mu}_i$. The decoder maps the resulting representation back to the text-feature space to reconstruct $\mathbf{x}_i$. This path encourages the latent representation to retain input information, while the posterior variance describes the spread of plausible representations around the mean.

The squared reconstruction error has a probabilistic interpretation when the decoder output is treated as the mean of a fixed-variance Gaussian observation model:

$$p_\psi(\mathbf{x}_i \mid \mathbf{z}_i) = \mathcal{N}\big(g_{\mathrm{dec}}(\mathbf{z}_i), \sigma_{\mathrm{obs}}^2 \mathbf{I}_{d_t}\big), \qquad \sigma_{\mathrm{obs}} > 0, \tag{38}$$

where ψ denotes the decoder parameters and σ_{obs} denotes a fixed observation standard deviation, so that σ_{obs}^2 is the fixed observation variance (Kingma & Welling, 2014; Blei et al., 2016). Under this model, squared reconstruction error is proportional to negative log likelihood up to an additive constant. The probabilistic model is therefore defined over frozen text embeddings rather than over sequences of text tokens.

B.2 From Reconstruction Error to the Variational Objective

The implemented reconstruction loss averages over both nodes and text-feature dimensions. Its expected form is:

$$\overline{\mathcal{L}}_{\mathrm{rec}} = \frac{1}{N d_t} \sum_{i=1}^{N} \mathbb{E}_{q_i}\left[\|\mathbf{x}_i - g_{\mathrm{dec}}(\mathbf{z}_i)\|_2^2\right]. \tag{39}$$

The single latent sample used during training provides a Monte Carlo estimate of this expectation. To express the same reconstruction term in the standard probabilistic form, define the node-averaged negative log likelihood and KL divergence as:

$$\overline{\mathrm{NLL}} = -\frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{q_i}[\log p_\psi(\mathbf{x}_i \mid \mathbf{z}_i)], \qquad \overline{D}_{\mathrm{KL}} = \frac{1}{N} \sum_{i=1}^{N} D_{\mathrm{KL}}(q_i \| p). \tag{40}$$

For one observation, the Gaussian model in Eq. (38) gives:

$$-\log p_\psi(\mathbf{x}_i \mid \mathbf{z}_i) = \frac{\|\mathbf{x}_i - g_{\mathrm{dec}}(\mathbf{z}_i)\|_2^2}{2\sigma_{\mathrm{obs}}^2} + \frac{d_t}{2} \log(2\pi\sigma_{\mathrm{obs}}^2). \tag{41}$$

Thus, averaging the squared error over $N d_t$ entries and averaging the negative log likelihood over N nodes introduce a factor of d_t. Similarly, the implemented KL loss is averaged over $N d_z$ entries, whereas $\overline{D}_{\mathrm{KL}}$ is averaged only over nodes and therefore introduces a factor of d_z. These two factors determine the effective balance between reconstruction and prior regularisation.

Normalised Variational Objective. For $\lambda_r > 0$ and $\lambda_{\mathrm{KL}} > 0$, define the implicit observation variance $\sigma_{\mathrm{impl}}^2 = \lambda_{\mathrm{KL}} d_t / (2\lambda_r d_z)$ and set $\sigma_{\mathrm{obs}}^2 = \sigma_{\mathrm{impl}}^2$ in Eq. (38). The expected auxiliary objective

then satisfies:

$$\lambda_r \overline{\mathcal{L}}_{\mathrm{rec}} + \lambda_{\mathrm{KL}} \mathcal{L}_{\mathrm{KL}} = \kappa_{\mathrm{VAE}} \left(\overline{\mathrm{NLL}} + \overline{D}_{\mathrm{KL}} \right) + c_0,$$
$$\kappa_{\mathrm{VAE}} = \frac{\lambda_{\mathrm{KL}}}{d_z}, \qquad c_0 = -\lambda_r \sigma_{\mathrm{impl}}^2 \log(2\pi\sigma_{\mathrm{impl}}^2). \tag{42}$$

where c_0 is independent of the encoder and decoder parameters.

Derivation. Equation (38), with $\sigma_{\mathrm{obs}}^2 = \sigma_{\mathrm{impl}}^2$, and the definition in Eq. (39) give:

$$\overline{\mathrm{NLL}} = \frac{d_t}{2\sigma_{\mathrm{impl}}^2} \overline{\mathcal{L}}_{\mathrm{rec}} + \frac{d_t}{2} \log(2\pi\sigma_{\mathrm{impl}}^2). \tag{43}$$

$$\overline{D}_{\mathrm{KL}} = \frac{1}{2N} \sum_{i=1}^{N} \sum_{r=1}^{d_z} \left(\mu_{i,r}^2 + \sigma_{i,r}^2 - \log \sigma_{i,r}^2 - 1 \right) = d_z \mathcal{L}_{\mathrm{KL}}. \tag{44}$$

The second identity is the closed-form KL divergence between a diagonal Gaussian and the standard Gaussian prior. The factor d_z appears because $\mathcal{L}_{\mathrm{KL}}$ averages the sum over both nodes and latent dimensions. Substituting these two identities into the weighted auxiliary loss yields Eq. (42).

B.3 Interpretation and Scope of the Evidence Bound

The standard variational identity writes the log evidence as the sum of a lower bound and the divergence between the approximate and exact posteriors (Blei et al., 2016):

$$\log p_\psi(\mathbf{x}_i) = \underbrace{\mathbb{E}_{q_i}[\log p_\psi(\mathbf{x}_i \mid \mathbf{z}_i)] - D_{\mathrm{KL}}(q_i \| p)}_{\text{evidence lower bound}} + D_{\mathrm{KL}}(q_i \| p_\psi(\mathbf{z}_i \mid \mathbf{x}_i)). \tag{45}$$

Nonnegativity of the posterior divergence, equivalently Jensen's inequality, then implies:

$$-\frac{1}{N} \sum_{i=1}^{N} \log p_\psi(\mathbf{x}_i) \le \overline{\mathrm{NLL}} + \overline{D}_{\mathrm{KL}}. \tag{46}$$

Combining Eqs. (42) and (46) shows that, under the implicit fixed-variance Gaussian observation model, the implemented auxiliary loss equals a positive scaling of the negative evidence lower bound plus a parameter-independent constant. Equivalently,

$$-\frac{\lambda_{\mathrm{KL}}}{d_z N} \sum_{i=1}^{N} \log p_\psi(\mathbf{x}_i) + c_0 \le \lambda_r \overline{\mathcal{L}}_{\mathrm{rec}} + \lambda_{\mathrm{KL}} \mathcal{L}_{\mathrm{KL}}. \tag{47}$$

Thus, after accounting for the different dimension normalisations, the weighted reconstruction and KL terms minimise an upper bound on the scaled average negative log evidence. In summary, this establishes a variational interpretation of the auxiliary semantic-encoding objective: the implemented loss is tied to the Gaussian encoder–decoder model through the implicit observation variance σ_{impl}^2 and the positive scale κ_{VAE}, both determined by λ_r and λ_{KL}.

C Theoretical Analysis of Semantic Graph Inference

The semantic graph is built by weighting a restricted set of candidate pairs. The following analysis explains the block component and the combined score conditional on the latent representations and node statistics, before graph symmetrisation and GCN normalisation.

C.1 Soft Memberships and Block Compatibility

Let $G_i \sim \mathrm{Categorical}(\boldsymbol{\pi}_i)$ and $G_j \sim \mathrm{Categorical}(\boldsymbol{\pi}_j)$ be independent latent assignments. Averaging the block-pair score over these assignments gives:

$$b_{i,j} = \boldsymbol{\pi}_i^\top \mathbf{B} \boldsymbol{\pi}_j = \sum_{c,c'} \pi_{i,c} \pi_{j,c'} B_{c,c'} = \mathbb{E}[B_{G_i,G_j}]. \tag{48}$$

This expectation resembles the averaging over interaction-specific memberships in mixed-membership block models (Airoldi et al., 2008). In the present formulation, however, $\mathbf{B}$ contains

real-valued scores rather than probabilities. Its entries need not lie in $[0, 1]$, and the sigmoid is applied only after the score components have been combined. Thus, the bilinear term is an expected block-pair score. Because the logistic function is nonlinear, $\mathrm{sigmoid}(\mathbb{E}[B_{G_i,G_j}])$ generally differs from $\mathbb{E}[\mathrm{sigmoid}(B_{G_i,G_j})]$. The resulting edge weight is consequently not, in general, the Bernoulli probability obtained by marginalising a classical SBM over latent assignments.

At the initialisation $\mathbf{B} = 2\mathbf{I}_K - \mathbf{1}\mathbf{1}^\top$, the score becomes:

$$b_{i,j} = 2\langle \boldsymbol{\pi}_i, \boldsymbol{\pi}_j \rangle - 1 \in [-1, 1]. \tag{49}$$

Concentrated memberships in the same block yield a high initial score, while diffuse or dissimilar memberships lower it. In particular, a positive score requires an inner product greater than $1/2$; sharing only the most probable block is insufficient. After training, Eq. (48) remains a convex combination of the entries of $\mathbf{B}$, but these learned entries need not remain within $[-1, 1]$.

C.2 Multiplicative Interpretation of Candidate-Edge Odds

With $d_{i,j} = \|\mathbf{z}_i - \mathbf{z}_j\|_2^2 / d_z$, the activity contribution in Eq. (23) is $\lambda_a \log(\theta_i\theta_j + \delta)$, and the logistic link yields:

$$\frac{a_{i,j}^{\mathrm{gen}}}{1 - a_{i,j}^{\mathrm{gen}}} = \exp(\ell_{i,j}) = e^{b_{i,j}} (\theta_i\theta_j + \delta)^{\lambda_a} e^{\alpha s_i^{\mathrm{info}} s_j^{\mathrm{info}}} e^{-\lambda_s (s_i^{\mathrm{struct}} + s_j^{\mathrm{struct}}) - \gamma d_{i,j}}. \tag{50}$$

The four multiplicative factors represent block compatibility, endpoint activity, membership ambiguity, and the joint penalties for posterior dispersion and latent distance, respectively. This factorisation also establishes the distinct effects of membership ambiguity and posterior dispersion: the former increases the odds through a bounded positive adjustment, whereas the latter decreases them. Latent distance supplies a separate pairwise penalty, so two nodes with similar block memberships can still receive a low edge weight when their latent representations are far apart.

C.3 Conditional Ranking and Edge Fitting

Since the logistic link is strictly increasing, the ordering of the candidate-edge weights equals the ordering of their logits. Holding all other score components fixed, increasing either positive endpoint activity increases the weight when $\lambda_a > 0$, whereas increasing posterior dispersion or latent distance decreases it when the corresponding penalty weight is positive.

The edge loss in Eq. (34) averages the observed-positive and sampled-negative binary cross-entropies separately. Under this objective, each sigmoid output parameterises a Bernoulli model for the edge label of a training pair. Applied to the restricted candidate set, Eq. (24) yields the continuous candidate-edge probability used in the main text. Its value combines block compatibility, membership ambiguity, endpoint activity, posterior dispersion, and latent distance through Eq. (23). SEGIA retains these probabilities as candidate-edge weights instead of thresholding them, preserving graded differences in modelled connectivity for semantic graph propagation. Symmetrisation and degree normalisation subsequently transform these weights for graph convolution.

D Analysis of Membership Ambiguity and Latent Uncertainty

SEGIA uses two node statistics that describe different properties of a semantic representation. Membership entropy measures ambiguity in the assignment to blocks, whereas posterior variance measures dispersion in the continuous latent space. The distinction between these statistics is specified before their joint contribution to edge inference is analysed.

D.1 Entropy of the Soft Block Memberships

For $K > 1$, the ideal normalised entropy is:

$$H_K(\boldsymbol{\pi}_i) = -\frac{1}{\log K} \sum_{c=1}^{K} \pi_{i,c} \log \pi_{i,c}, \qquad 0 \le H_K(\boldsymbol{\pi}_i) \le 1, \tag{51}$$

with $0 \log 0 = 0$. It is zero for a one-hot membership and reaches one for the uniform distribution. The implemented statistic s_i^{info} replaces the logarithm argument by $\max(\pi_{i,c}, 10^{-8})$ and clips the result to $[0, 1]$. Consequently, for $\alpha \geq 0$,

$$0 \leq \alpha s_i^{\text{info}} s_j^{\text{info}} \leq \alpha, \qquad 1 \leq e^{\alpha s_i^{\text{info}} s_j^{\text{info}}} \leq e^{\alpha}. \tag{52}$$

The term is a bounded diffuse-membership adjustment, strongest when both endpoints have diffuse memberships. If either endpoint has an ideal one-hot membership, the product vanishes even when the other endpoint is diffuse; if both are uniform, the product reaches its maximum. The adjustment is designed to help candidate pairs involving two broadly scoped or cross-topic nodes whose semantics span several coherent groups. For such nodes, the bilinear block score distributes membership mass over many block pairs and can dilute each individually compatible contribution, even when the two nodes remain close in the latent space. The positive term partially compensates for this dilution when both endpoints exhibit diffuse memberships. It is therefore most relevant to candidate pairs for which both nodes have multi-group semantic support, rather than pairs containing only one ambiguous endpoint. Requiring both endpoints to be diffuse keeps the adjustment pair-specific, because a single high-entropy node cannot raise all of its incident candidate scores through this term alone. Entropy measures concentration rather than agreement, so block-level agreement remains represented by $b_{i,j}$. Endpoint activity, posterior dispersion, and latent distance continue to constrain the final score. The term therefore provides a bounded compensation for multi-group semantic relations without independently establishing that a candidate pair should receive a large edge weight.

D.2 Posterior Variance as Latent Dispersion

For the diagonal Gaussian encoder, the second statistic has the exact identity:

$$s_i^{\text{struct}} = \frac{1}{d_z} \sum_{r=1}^{d_z} \sigma_{i,r}^2 = \frac{1}{d_z} \mathbb{E}_{q_i} \left[\|\mathbf{z}_i - \boldsymbol{\mu}_i\|_2^2 \right]. \tag{53}$$

It is the expected squared deviation from the posterior mean per latent dimension. A larger value therefore corresponds to a broader approximate posterior and greater uncertainty about the precise latent semantic position of the node. Although the notation s_i^{struct} is retained from the main text, the statistic is computed from the semantic encoder and is used to quantify uncertainty in the inferred semantic representation. Clipping the log variance to $[-10, 10]$ implies $e^{-10} \leq s_i^{\text{struct}} \leq e^{10}$.

The distinction from pairwise distance can also be expressed under independent posterior draws for two distinct nodes:

$$\frac{1}{d_z} \mathbb{E}_{q_i q_j} \left[\|\mathbf{z}_i - \mathbf{z}_j\|_2^2 \right] = \frac{\|\boldsymbol{\mu}_i - \boldsymbol{\mu}_j\|_2^2}{d_z} + s_i^{\text{struct}} + s_j^{\text{struct}}. \tag{54}$$

The identity follows by expanding the square: the centred cross terms vanish, leaving the squared mean separation and the two covariance traces. This decomposition establishes the complementary roles of posterior dispersion and pairwise semantic distance. The distance between posterior means measures separation between two nodes, whereas the two dispersion terms measure spread around their respective latent positions. Under the model's scoring assumption, a broader posterior provides less concentrated latent support for a precise semantic relation. The scorer therefore uses endpoint dispersion as a conservative negative adjustment, without treating it as a calibrated estimate of edge error.

D.3 Joint Effect on Candidate-Edge Weights

To expose how the two statistics modify an otherwise fixed candidate-edge score, define the baseline logit and its corresponding weight as:

$$\ell_{i,j}^{(0)} = b_{i,j} + \lambda_a \log(\theta_i \theta_j + \delta) - \gamma d_{i,j}, \qquad a_{i,j}^{(0)} = \text{sigmoid}(\ell_{i,j}^{(0)}). \tag{55}$$

The full logit is $\ell_{i,j} = \ell_{i,j}^{(0)} + u_{i,j}$, where:

$$u_{i,j} = \alpha s_i^{\text{info}} s_j^{\text{info}} - \lambda_s (s_i^{\text{struct}} + s_j^{\text{struct}}).$$

Consequently, the combined effect of membership ambiguity and posterior dispersion has the exact odds-ratio form:

$$\frac{a_{i,j}^{\text{gen}}/(1-a_{i,j}^{\text{gen}})}{a_{i,j}^{(0)}/(1-a_{i,j}^{(0)})} = \exp\big(\alpha s_i^{\text{info}} s_j^{\text{info}} - \lambda_s(s_i^{\text{struct}} + s_j^{\text{struct}})\big)\,. \tag{56}$$

Thus, the two terms compete through a single net adjustment. Relative to the baseline score, the odds increase when $\alpha s_i^{\text{info}} s_j^{\text{info}} > \lambda_s(s_i^{\text{struct}} + s_j^{\text{struct}})$, remain unchanged when the two sides are equal, and decrease when the dispersion penalty dominates. Even a positive adjustment does not by itself imply a large final edge weight, because the baseline still contains block compatibility, endpoint activity, and latent distance. This decomposition describes the intended roles of the two statistics: membership ambiguity compensates for attenuation of multi-group candidate pairs, whereas posterior dispersion conservatively reduces the influence of relations whose latent support is less concentrated.

E Additional Experimental Results

This section examines the effects of cross-view alignment, variational regularisation, edge-score penalties, and candidate budgets on node classification. Each figure reports the observed response of every benchmark separately.

E.1 Sensitivity to Cross-View Alignment

The coefficient λ_o controls the contribution of prediction alignment, while τ_o controls the softness of the distributions being aligned. Figure 3 evaluates $\lambda_o \in \{0.1, 0.5, 1.0\}$ and $\tau_o \in \{0.2, 0.5, 1.0\}$.

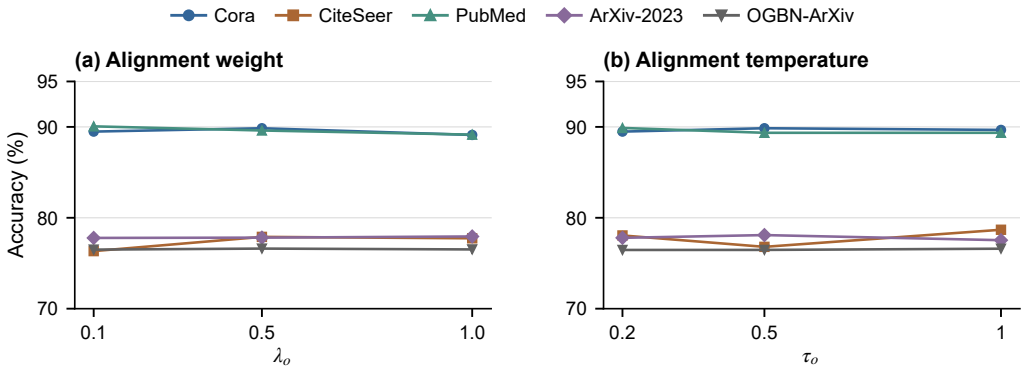

Figure 3: Sensitivity to alignment weight λ_o and temperature τ_o, where the horizontal axes show all three tested values.

As illustrated, neither alignment parameter produces a shared monotonic trend across all benchmarks. A moderate alignment weight gives the strongest observed result on most datasets, whereas individual datasets prefer a weaker or stronger coupling. This pattern suggests that alignment is most useful as a balanced constraint rather than an increasingly dominant objective. The temperature curves are generally stable, although the preferred degree of smoothing remains dataset dependent. Together, the results support moderate default settings while showing that neither parameter has a universally optimal value.

E.2 Sensitivity to Variational Regularisation

The KL coefficient controls how strongly the approximate posterior is regularised towards the prior. As derived in Appendix B, its effective balance with reconstruction also depends on feature and latent dimensions. Figure 4 evaluates $\lambda_{\text{KL}} \in \{10^{-3}, 10^{-2}, 5 \times 10^{-2}\}$.

As illustrated, performance is largely stable across the tested KL weights, with only one dataset showing a pronounced interior optimum. A very weak KL term provides limited prior regulari-

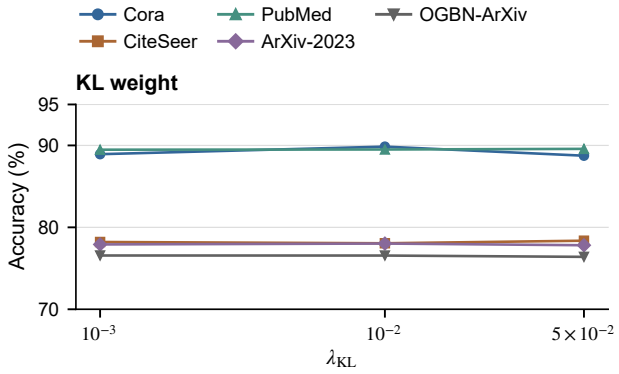

Figure 4: Sensitivity to the KL weight, where the horizontal axis is logarithmic and shows all three tested values.

sation, whereas a stronger term can restrict the information retained by the latent representation. The intermediate setting therefore provides a conservative balance between reconstruction fidelity and posterior regularisation. Because the remaining curves are nearly flat, the experiment mainly indicates robustness within the tested range rather than a universal optimum.

E.3 Sensitivity to Dispersion and Distance Penalties

The coefficient γ penalises pairwise latent distance, whereas λ_s penalises endpoint posterior dispersion. Both are evaluated at $\{0.25, 0.5, 1.0\}$ in Figure 5.

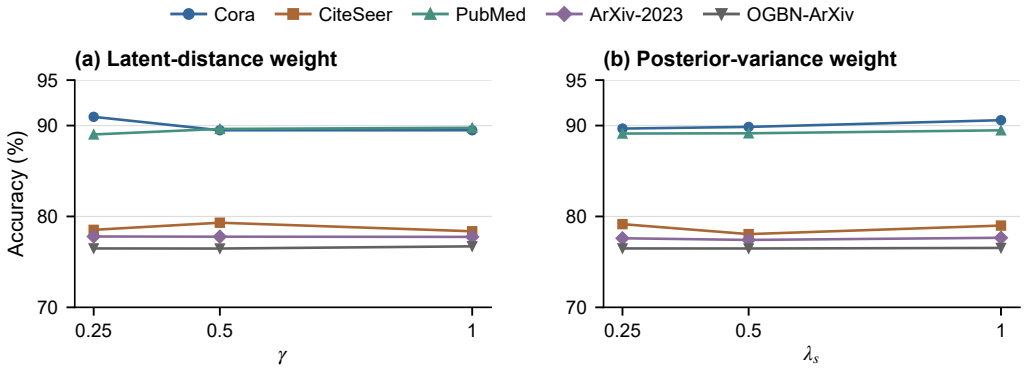

Figure 5: Sensitivity to latent-distance weight γ and posterior-dispersion weight λ_s.

As shown, the latent-distance penalty has no consistent preferred strength, that is, some datasets benefit from a weaker penalty, whereas others remain stable or improve as the penalty increases. This indicates that latent distance acts as a dataset-dependent filter rather than a uniformly stronger source of supervision. The posterior-dispersion penalty shows a clearer tendency, with the strongest tested value performing best on most datasets. This pattern is consistent with suppressing candidate relations supported by diffuse latent representations, although the exception on CiteSeer shows that excessive suppression can remove useful relations. Overall, moderate penalty values provide a stable default, while the optimal balance depends on the reliability of the learned semantic space.

E.4 Sensitivity to Candidate-Neighbour Budgets

The text-neighbour and two-hop budgets determine which additional pairs enter the candidate graph before edge scoring. The three tested values $\{3, 5, 10\}$ are treated as discrete graph-construction configurations while the remaining candidate sources are retained. Figure 6 uses grouped bars to

compare the measured settings within each dataset and shows that increasing the number of available pairs does not uniformly improve classification.

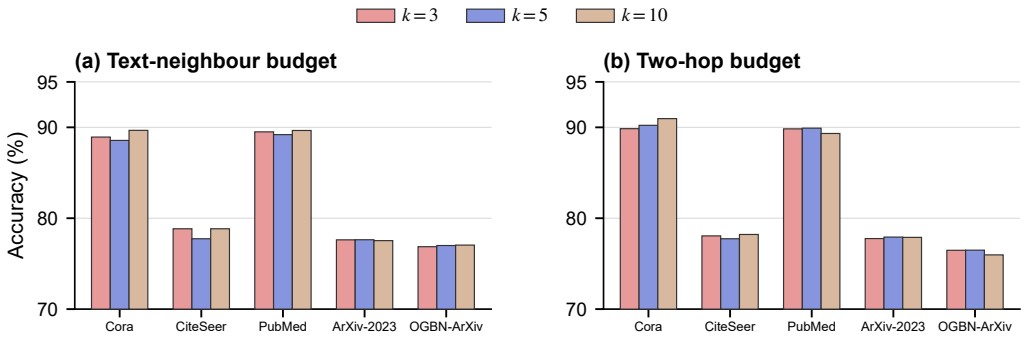

Figure 6: Sensitivity to candidate-neighbour budgets, with observed edges and the other candidate source retained while the displayed budget varies.

As illustrated, enlarging the text-neighbour budget does not yield a uniform monotonic improvement. A wider semantic neighbourhood is useful on several datasets, but others remain stable, indicating that candidate quality matters more than candidate count alone. The two-hop budget is even more dataset dependent: a larger budget benefits Cora, while several other datasets plateau or decline as more structurally reachable pairs are added. These trends suggest that overly small budgets can omit useful relations, whereas overly large budgets can introduce redundant candidates and additional computation. Overall, the budgets should be treated as dataset-specific sparsity controls rather than quantities that should be increased monotonically.

F Datasets and Implementation Details

F.1 Dataset description

We conduct experiments on five TAGs, namely Cora (Sen et al., 2008), CiteSeer (Giles et al., 1998), PubMed (Sen et al., 2008), ArXiv-2023 (He et al., 2024), and OGBN-ArXiv (He et al., 2024; Hu et al., 2020).

All five benchmarks are citation networks in which nodes represent scientific publications and edges represent citation relations. Specifically, Cora and CiteSeer cover seven machine-learning categories and six computer-science areas, respectively, whereas PubMed contains diabetes-related publications from three categories. ArXiv-2023 and OGBN-ArXiv use the same 40 arXiv computer-science subject categories but differ substantially in graph scale and data provenance. The experiments use the 3,186 node real-text variant of CiteSeer. Dataset statistics and label vocabularies are reported in Tables 3 and 4, respectively.

Table 3: Statistics of the evaluation datasets.

Datasets	**#Classes**	**#Nodes**	**#Edges**
Cora	7	2,708	5,429
CiteSeer	6	3,186	4,277
PubMed	3	19,717	44,338
ArXiv-2023	40	46,198	78,543
OGBN-ArXiv	40	169,343	1,166,243

Table 4: Label Vocabularies used by the benchmark tasks and text-expansion prompts.

Datasets	Label Vocabulary
Cora	Case Based, Genetic Algorithms, Neural Networks, Probabilistic Methods, Reinforcement Learning, Rule Learning, Theory.
CiteSeer	Agents, Artificial Intelligence, Databases, Information Retrieval, Machine Learning, Human–Computer Interaction.
PubMed	Type 1 diabetes, Type 2 diabetes, Experimentally induced diabetes.
ArXiv-2023; OGBN-ArXiv	cs.AI, cs.AR, cs.CC, cs.CE, cs.CG, cs.CL, cs.CR, cs.CV, cs.CY, cs.DB, cs.DC, cs.DL, cs.DM, cs.DS, cs.ET, cs.FL, cs.GL, cs.GR, cs.GT, cs.HC, cs.IR, cs.IT, cs.LG, cs.LO, cs.MA, cs.MM, cs.MS, cs.NA, cs.NE, cs.NI, cs.OH, cs.OS, cs.PF, cs.PL, cs.RO, cs.SC, cs.SD, cs.SE, cs.SI, cs.SY.

F.2 Preprocessing and Evaluation Protocol

For a consistent evaluation, all datasets are randomly partitioned into training, validation, and test sets in a 6:2:2 ratio. The split masks are determined by the run seed, and validation accuracy is used to select the checkpoint before test evaluation.

F.3 Implementation Details

SEGIA encodes text augmented with Qwen2.5-7B-Instruct responses using the frozen all-MiniLM-L6-v2 encoder. Two independent two-layer GCNs process the observed and semantic views, and the soft-membership dimension equals the number of classes. We train with Adam for 200 epochs and select the checkpoint by validation accuracy. The numerical values below describe a reference configuration, not a shared setting imposed on every dataset.

This configuration uses a learning rate of 0.01, weight decay of 10^{-3}, dropout of 0.5, and hidden and latent dimensions of 128. The objective weights are $\lambda_{\mathrm{KL}} = 10^{-2}$, $\lambda_r = 0.1$, $\lambda_e = 0.5$, $\lambda_v = 0.5$, and $\lambda_o = 0.5$ for the KL, reconstruction, edge-BCE, view-classification, and cross-view OT terms, respectively.

For continuous-SBM scoring, the membership-ambiguity coefficient is $\alpha = 1.0$, the node-activity coefficient is $\lambda_a = 0.5$, the posterior-dispersion coefficient is $\lambda_s = 0.5$, and the latent-distance coefficient is $\gamma = 0.5$. The soft-membership temperature is 1.0. Candidate pairs come from observed edges, text-embedding nearest neighbours, and two-hop neighbours. Each node contributes up to five text neighbours and five two-hop neighbours.

For entropic OT, the alignment temperature is 0.5, the regularisation coefficient is 0.1, and the number of Sinkhorn iterations is 20. Response-augmented text is enabled. The reported results use dataset-specific run configurations, including the corresponding response files and cached sentence embeddings; these configurations will be released with the code.

G Text Expansion Prompts and Example

G.1 Text Expansion Procedure

Text expansion supplies the language model with a node's document text and a task-level label vocabulary, then requests ranked categories and supporting explanations. No node-specific ground-truth class is supplied. The generated response is used as an additional textual feature, rather than as a ground-truth training label. The scripts use Qwen2.5-7B-Instruct with deterministic decoding.

G.2 Dataset-Specific Instructions

Table 5 transcribes the script instructions, normalising whitespace and retaining the repeated Cora sentence. The input wrapper is "Abstract: {abstract}" then "Title: {title}" for Cora, PubMed, and both arXiv datasets, or "Document: {text}" for CiteSeer. It is followed by "Question:", the listed instruction, and the generation cue "Answer:".

Table 5: Prompts used in this work to query the LLM.

Datasets	Question Instruction
Cora	Which of the following sub-categories of AI does this paper belong to: Case Based, Genetic Algorithms, Neural Networks, Probabilistic Methods, Reinforcement Learning, Rule Learning, Theory? If multiple options apply, provide a comma-separated list ordered from most to least related. Only choose categories that are explicitly supported by the title or abstract. Do not include a category based on loose analogy, common association, or methods that might be used but are not mentioned. Only explain the categories you choose. Only explain the categories you choose. Do not mention, list, compare, or discuss any categories you do not choose. Do not write sentences about other categories being unsupported, not mentioned, not implied, irrelevant, or not applicable. Then, for each choice you gave, explain how it is present in the text.
CiteSeer	Which of the following Citeseer research categories does this paper belong to: Agents, Artificial Intelligence, Databases, Information Retrieval, Machine Learning, Human-Computer Interaction? Select at least one category. If multiple categories apply, provide a comma-separated list ordered from most to least related. Only choose categories directly supported by an explicit statement, task, method, or technical concept in the document. For each chosen category, include a short exact supporting phrase from the document and explain how that phrase supports the category. Do not select a category based only on field hierarchy, common association, or a method that is merely often used in that field. Do not justify a category with indirect, speculative, or external reasoning. Only explain the categories you choose. Do not mention, list, compare, or discuss categories you do not choose.
PubMed	Does the paper involve any cases of Type 1 diabetes, Type 2 diabetes, or Experimentally induced diabetes? Please give one or more answers of either Type 1 diabetes, Type 2 diabetes, or Experimentally induced diabetes; if multiple options apply, provide a comma-separated list ordered from most to least related, then for each choice you gave, give a detailed explanation with quotes from the text explaining why it is related to the chosen option.
ArXiv-2023	Which arXiv CS subcategory does this paper belong to? Give 5 likely arXiv CS subcategories as a comma-separated list ordered from most to least likely, in the form "cs.XX", and provide your reasoning. Rank categories by how directly they describe the paper's central research problem and contribution, rather than merely a method, tool, or possible application. For each category, use direct evidence from the title or abstract and avoid speculative reasoning. Predict the category an author would most likely choose as the paper's primary arXiv subject, rather than every field related to its methods or applications. Rank categories that directly describe the paper's main research object or task above broader categories that describe only methods, tools, or possible applications.
OGBN-ArXiv	Which arXiv CS subcategory does this paper belong to? Give 5 likely arXiv CS subcategories as a comma-separated list ordered from most to least likely, in the form "cs.XX", and provide your reasoning. Rank categories by how directly they describe the paper's central research problem and contribution, rather than merely a method, tool, or possible application. For each category, use direct evidence from the title or abstract and avoid speculative reasoning. Predict the category an author would most likely choose as the paper's primary arXiv subject, rather than every field related to its methods or applications. Rank categories that directly describe the paper's main research object or task above broader categories that describe only methods, tools, or possible applications.

G.3 ILLUSTRATIVE TEXT EXPANSION

Table 6 presents the stored response for Cora node 0. The response was generated by the local Qwen2.5-7B-Instruct model used for Cora text expansion and identifies the document with the category Probabilistic Methods.

Table 6: Example of a stored Cora text expansion.

Fields	**Content**
Title	The megaprior heuristic for discovering protein sequence patterns.
Input excerpt	"This paper presents a solution to the problem of convex combinations in the form of a heuristic based on using extremely low variance Dirichlet mixture priors as part of the statistical model."
Predicted category	Probabilistic Methods.
Stored response	"Probabilistic Methods Explanation: The paper explicitly discusses the use of statistical models and Dirichlet mixture priors, which fall under the category of Probabilistic Methods. Specifically, the abstract mentions 'fitting the parameters of a statistical model to a group of related sequences' and describes a solution involving 'extremely low variance Dirichlet mixture priors.' These elements clearly indicate the application of probabilistic methods in the context of analyzing protein sequence patterns."