T-GINEE: A Tensor-Based Multilayer Graph Representation Learning

Anonymous Authors[1]

Abstract

While traditional network analysis focuses on single-layer networks, real-world systems often form multilayer networks with multiple relationship types. However, existing methods typically fail to capture complex inter-layer dependencies by treating layers independently or aggregating them. To address this, we propose T-GINEE (Tensor-Based Generalized Multilayer-graph Estimating Equation), a statistical regularization framework combining tensor-based generalized estimating equations with task-specific loss to model cross-network correlations explicitly. Key innovations include: (1) CP tensor decomposition capturing structural dependencies via shared latent factors; (2) a generalized estimating equation framework modeling inter-layer correlations through working covariance matrices; and (3) a flexible link function accommodating characteristics like sparsity. Our theoretical analysis establishes consistency and asymptotic normality under mild conditions. Extensive experiments on synthetic and real-world datasets validate T-GINEE's effectiveness for multilayer network analysis. Our code is available in the **supplementary materials** to ensure reproducibility.

1. Introduction

In the real world, interactions between entities are often multifaceted, with these multi-relational characteristics engaging one another under varied circumstances or through distinct modalities. For instance, in social networks (Van Den Oord & Van Rossem, 2002), individuals may be connected through multiple relationship types such as friends, colleagues, and family. In biology (Zheng et al., 2019), genes or proteins exhibit various collaboration schemes like co-expression and physical interactions. In global trade, countries exchange a wide range of different commodities.

For such intricate relational landscapes, a multi-layer graph offers a faithful and structured representation. This architecture is defined by a common set of vertices, where each layer is endowed with a unique edge set to delineate a specific type of relation. Such graphs are prevalent across numerous disciplines, including social graphs that capture multiple interaction channels between individuals (Greene & Cunningham, 2013), biological graphs detailing different collaboration schemes among genes or proteins (Li et al., 2020; Liu et al., 2020), and global trade graphs mapping the exchange of various commodities (Alves et al., 2019; Ren et al., 2020). To effectively analyze these intricate structures, a fundamental step is to learn low-dimensional vector representations (i.e., embeddings) for the entities that capture the complex relational information encoded across layers.

Numerous approaches have been developed for graph embedding, employing various techniques such as similarity indices (Boden et al., 2017), maximum likelihood models (Yuan & Qu, 2021), matrix factorization (Tang et al., 2009; Dong et al., 2012; Gligorijević et al., 2016), and graph neural networks (Kipf & Welling, 2016; Hamilton et al., 2017; Xu et al., 2018). For multilayer graph embedding, which provides a richer representation of complex systems (Kivelä et al., 2014), analysis often involves extending these single-layer techniques. Prominent approaches include tensor-based methods that leverage the natural tensor structure of multilayer graphs (Kolda & Bader, 2009; Aguiar et al., 2024), as well as adaptations of deep learning models like GCNs and random-walk embeddings (Ghorbani et al., 2019; Song & Thiagarajan, 2018).

However, a critical challenge underlying many of these methods is the lack of a rigorous theoretical foundation for the multi-layer context. While embedding learning has proven effective for single-layer graphs (Cai et al., 2018), we lack robust theoretical frameworks systematically characterizing the embedding process across multiple layers (Interdonato et al., 2020). This absence of formal tools describing how embeddings capture and preserve cross-layer dynamics significantly impedes developing principled approaches, representing a fundamental limitation in the field (Shanthamallu et al., 2019; Jiao et al., 2021; Lyu et al., 2023).

Without this theoretical guidance, existing approaches often

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

resort to simplistic solutions, such as learning representations for layers independently (Tang et al., 2009; Dong et al., 2012) or using basic aggregation techniques (Paul & Chen, 2020; Lei et al., 2020). These methods lack the grounding to explain how embeddings should optimally encode the nuanced ways in which relationships in one layer might influence or contradict another (Liu et al., 2017; Zhang et al., 2018). This deficit is especially problematic for real-world systems where entities engage through multiple relation types simultaneously (Xu et al., 2020; Yang et al., 2020), highlighting the urgent need for new frameworks that can faithfully represent this complex interplay (Huang et al., 2020; Shanthamallu et al., 2019).

To address these challenges, we propose T-GINEE (Tensor-based Generalized Multilayer-graph Estimating Equation), a statistical regularization framework that combines tensor-based generalized estimating equations with task-specific loss to explicitly model cross-network correlations. The key technical innovations of T-GINEE include: (1) A CP tensor decomposition approach that effectively captures structural dependencies through shared latent factors while maintaining computational efficiency; (2) A generalized estimating equation framework that explicitly models the correlations between different network layers through working covariance matrices; and (3) A flexible link function design that accommodates various network characteristics, including sparsity. Unlike previous approaches that rely on simple aggregation or separate modeling (Paul & Chen, 2020; Tang et al., 2009; Lei et al., 2020), T-GINEE provides a principled statistical framework to jointly model multiple networks.

Overall, T-GINEE integrates a symmetric CP tensor decomposition with a generalized estimating equation (GEE) formulation, provides asymptotic statistical guarantees, and validates the framework on both synthetic and real-world multilayer networks. The main contributions are summarized as follows:

- **Tensor-based Statistical Framework**: We propose a regularization framework combining tensor CP decomposition with generalized estimating equations. This approach explicitly models cross-network dependencies via a principled formulation while ensuring tractability.
- **Theoretical Guarantees**: We establish T-GINEE's consistency and asymptotic normality under mild conditions. The framework offers provable guarantees for estimation accuracy and convergence through rigorous analysis of the tensor-based estimating equations.
- **Empirical Validation**: Comprehensive experiments on synthetic and real-world networks demonstrate T-GINEE's effectiveness.

2. Methodology

In this section, we present Tensor-based Generalized Estimating Equations (T-GINEE), a framework for learning embeddings from multilayer graphs. Our method combines a low-rank CP parameterization of a multilayer graph with a generalized estimating equations (GEE) estimator, equipped with a structured working covariance, to jointly model within-layer and cross-layer dependencies.

2.1. Overview

Real-world networks often exhibit complex interdependencies, where multiple network structures coexist and influence each other. For instance, an individual's friendship networks on multiple social media platforms (such as Facebook, LinkedIn, and TikTok) form correlated multilayer graphs over the same set of users. We propose a statistical regularization framework that leverages tensor-based generalized estimating equations to explicitly model such cross-network correlations. Our framework, referred to as T-GINEE, consists of several core components illustrated in Figure 1: (i) a symmetric CP decomposition of the parameter tensor Θ of the multilayer graph into node embeddings α and layer-specific embeddings β; (ii) the construction of a parameter vector γ from these embeddings; and (iii) a tensor-based GEE formulation that estimates γ under a working covariance model and thereby captures complex dependencies across layers.

2.2. Problem formulation

Consider a multilayer network/graph $\mathcal{G} = \big(\mathcal{V}, \{\mathcal{G}^{(m)}\}_{m=1}^{M}\big)$,, where $\mathcal{V} = \{v_1, \ldots, v_n\}$ is a common set of vertices that interact across M different but potentially correlated network layers. Each layer $\mathcal{G}^{(m)} = (\mathcal{V}, \mathcal{E}^{(m)})$ captures a distinct type of relationship. We focus on undirected networks and index undirected edges by unordered pairs $\{i, j\}$, using the convention that $i \leq j$. In our notation we include pairs with $i = j$; in the empirical datasets we consider, the diagonal entries are identically zero (i.e., $\mathcal{A}_{i,i,m} \equiv 0$), so allowing $i = j$ does not affect estimation but simplifies asymptotic analysis.

Let $\mathcal{A} \in \{0,1\}^{n \times n \times M}$ be the adjacency tensor, where $\mathcal{A}_{i,j,m} = 1$ indicates an edge of type m between nodes i and j, and $\mathcal{A}_{i,j,m} = 0$ otherwise. For each pair (i, j) with $i \leq j$, the vector $\mathcal{A}_{i,j,\cdot} \in \mathbb{R}^M$ is treated as an M-dimensional binary response with mean $\mathcal{P}_{i,j,\cdot}(\Theta)$ and covariance matrix $\Sigma_{i,j}$. In line with the GEE paradigm, we specify only its first two moments, that is $\mathbb{E}[\mathcal{A}_{i,j,\cdot}] = \mathcal{P}_{i,j,\cdot}(\Theta)$ and $\mathrm{Cov}(\mathcal{A}_{i,j,\cdot}) = \Sigma_{i,j}$, without imposing a full joint distribution. Conditional on the parameters, we assume that the edge vectors $\{\mathcal{A}_{i,j,\cdot} : i \leq j\}$ are independent across different node pairs (i, j). Within each pair (i, j), the components

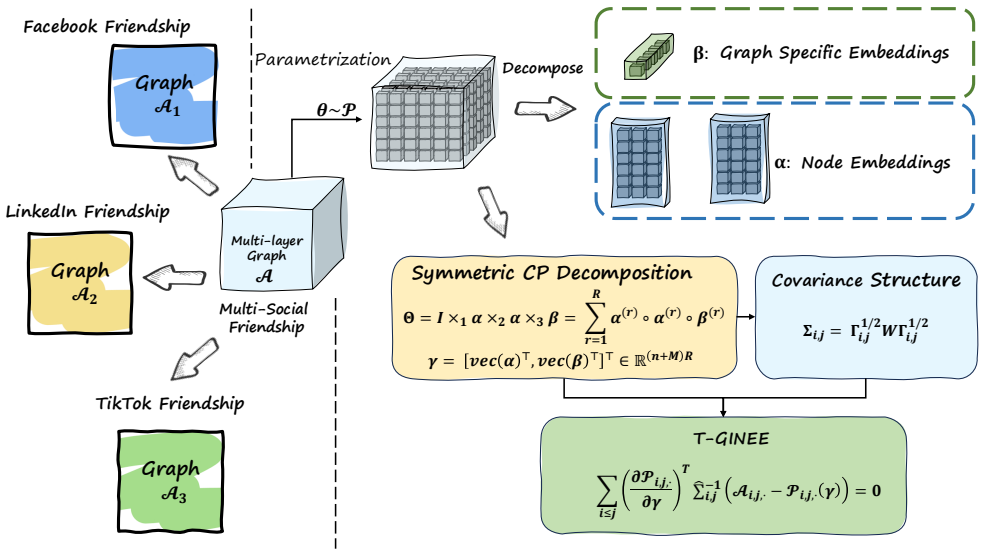


Figure 1. Schematic overview of the T-GINEE framework. Given a multilayer adjacency tensor $\mathcal{A}$, we introduce a parameter tensor Θ and perform a symmetric CP decomposition to obtain node embeddings α and layer-specific embeddings β. These embeddings are concatenated into a parameter vector γ. By solving tensor-based generalized estimating equations (T-GINEE) under a working covariance structure, the model learns γ and captures cross-layer dependencies in multilayer social networks (e.g., Facebook, LinkedIn, TikTok).

$\mathcal{A}_{i,j,1}, \ldots, \mathcal{A}_{i,j,M}$ may be correlated and this dependence is captured by $\Sigma_{i,j}$.

The parameter tensor $\Theta \in \mathbb{R}^{n \times n \times M}$ is linked to the mean tensor $\mathcal{P}$ through a known three-times continuously differentiable link function g, applied elementwise, such that

$$\mathcal{P}_{i,j,\cdot} = g^{-1}(\Theta_{i,j,\cdot}).$$

Typical choices of the link function include:

- the identity link $g(x) = x$, primarily used for weighted networks where $\mathcal{A}_{i,j,m}$ need not be binary; for Bernoulli observations, the identity link can be applied by implicitly constraining $\Theta_{i,j,m} \in [0, 1]$;
- the probit link $g(x) = \Phi^{-1}(x)$ with inverse $g^{-1}(x) = \Phi(x) = \int_{-\infty}^{x} (2\pi)^{-1/2} \exp(-t^2/2)\, \mathrm{d}t$;
- the logit link $g(x) = \log\{x/(1-x)\}$ with inverse $g^{-1}(x) = 1/(1+e^{-x})$;
- a sparsity-aware logit $g(x) = \log\{x/(s-x)\}$ with inverse $g^{-1}(x) = \dfrac{s}{1+e^{-x}}$, where $0 < s < 1$ is a sparsity coefficient that can decrease with n and M to accommodate increasingly sparse networks. When s is allowed to depend on (n, M), we assume there exists $\varepsilon > 0$ such that, for all (i, j, m) and all γ in a neighborhood of γ_0,

$$\varepsilon \le \mathcal{P}_{i,j,m}(\gamma) \le s - \varepsilon,$$

which in particular guarantees that g' and the required higher-order derivatives remain uniformly bounded, in line with Assumption 3.1.

2.3. Low-rank tensor decomposition

Before introducing the formal CP decomposition, we briefly motivate this modeling choice. In many multilayer systems, the same set of nodes participates in different relation types (layers) through a shared, low-dimensional set of latent traits (e.g., social roles, functional modules, or economic profiles). A symmetric CP factorization of the parameter tensor Θ reflects this idea: the node embeddings α define a common latent space across all layers, while the layer embeddings β determine how each relation type weights these latent factors. This yields a compact representation that captures higher-order interactions across nodes and layers, reduces the number of free parameters, and aligns with empirical findings that a relatively small number of latent dimensions often suffices to explain multilayer network structure.

In our asymptotic analysis, we consider a regime where n grows while M and R are either fixed or increase sufficiently slowly, more precisely satisfying $(n+M)R = o(n^{1/3})$ (see Assumption 3.2). To effectively model structural dependencies while maintaining computational efficiency, we employ a symmetric CP decomposition for the parameter tensor Θ:

$$\Theta = \mathcal{I} \times_1 \alpha \times_2 \alpha \times_3 \beta = \sum_{r=1}^{R} \alpha^{(r)} \circ \alpha^{(r)} \circ \beta^{(r)}, \quad (1)$$

where $\mathcal{I} \in \mathbb{R}^{R \times R \times R}$ denotes the order-3 identity tensor used in CP decomposition, $\alpha \in \mathbb{R}^{n \times R}$ contains node embeddings and $\beta \in \mathbb{R}^{M \times R}$ contains layer-specific embeddings. Here $\alpha^{(r)}$ and $\beta^{(r)}$ denote the r-th columns of α and β, respectively, and $\circ$ denotes the outer product. This

parameterization enforces $\Theta_{i,j,m} = \Theta_{j,i,m}$ for all i, j, m, consistent with undirected layers, and inherently imposes structural constraints through shared latent factors.

For optimization purposes, we vectorize these factor matrices into a compact representation:

$$\gamma = \left[\mathrm{vec}(\alpha)^\top,\ \mathrm{vec}(\beta)^\top\right]^\top \in \mathbb{R}^{(n+M)R}. \tag{2}$$

This parameter vector γ encapsulates the essential cross-layer dependencies and serves as the decision variable in our estimating equations. The low-rank tensor decomposition offers several advantages: it significantly reduces the number of free parameters, improving computational efficiency and mitigating overfitting; by sharing latent factors across dimensions, it naturally captures the inherent relationships between nodes and layers; and it yields interpretable components, where α represents node-level patterns and β captures layer-level characteristics.

2.4. Tensor-based statistical regularization

We now introduce the tensor-based generalized estimating equations that define T-GINEE. A detailed derivation is provided in **Appendix** A. The tensor-based estimating equations for multilayer graphs are

$$\sum_{i\le j}\left(\frac{\partial \mathcal{P}_{i,j,\cdot}}{\partial \gamma}\right)^\top \widehat{\Sigma}_{i,j}^{-1}\left(\mathcal{A}_{i,j,\cdot} - \mathcal{P}_{i,j,\cdot}(\gamma)\right) = \mathbf{0}, \tag{3}$$

where $\widehat{\Sigma}_{i,j}$ is a working covariance matrix. We denote the left-hand side of (3) by $s(\gamma)$; thus T-GINEE seeks a root $\hat{\gamma}$ of $s(\gamma) = \mathbf{0}$.

To compute the score contributions, we first derive $\partial \mathcal{P}_{i,j,\cdot}/\partial\gamma$ via the chain rule: starting from the Jacobian $\partial\,\mathrm{vec}(\Theta)/\partial\gamma$ implied by the CP decomposition, then applying the derivative of the link function, and finally projecting onto the (i,j)-th fibers using basis tensors $\mathcal{E}^{(i,j,m)}$.

Specifically, since $\mathcal{P}_{i,j,m} = g^{-1}(\Theta_{i,j,m})$ and g is differentiable, we have

$$\frac{\partial \mathcal{P}_{i,j,\cdot}}{\partial \gamma} = \left[\mathrm{diag}\big(g'(\mathcal{P}_{i,j,1}), \ldots, g'(\mathcal{P}_{i,j,M})\big)\right]^{-1} \frac{\partial \Theta_{i,j,\cdot}}{\partial \gamma} \tag{4}$$

where g' denotes the derivative of g, applied elementwise, and the diagonal matrix has (m,m)-th entry $g'(\mathcal{P}_{i,j,m})$ for $m \in [M]$. Let $\mathcal{E}^{(i,j,m)} \in \mathbb{R}^{n\times n\times M}$ be the tensor unit whose (i',j',m')-th entry is $\mathbf{1}\{(i,j,m) = (i',j',m')\}$. Then

$$\frac{\partial \Theta_{i,j,\cdot}}{\partial \gamma} = \left[\mathrm{vec}(\mathcal{E}^{(i,j,1)}), \ldots, \mathrm{vec}(\mathcal{E}^{(i,j,M)})\right]^\top \frac{\partial\,\mathrm{vec}(\Theta)}{\partial \gamma} \tag{5}$$

Under the CP decomposition of Θ, the Jacobian matrix with respect to the parameter vector γ takes the block form

$$\frac{\partial\,\mathrm{vec}(\Theta)}{\partial \gamma} = \begin{bmatrix} (\beta^{(1)})^\top \otimes \big(I_n \otimes \alpha^{(1)} + \alpha^{(1)} \otimes I_n\big) \\ \vdots \\ (\beta^{(R)})^\top \otimes \big(I_n \otimes \alpha^{(R)} + \alpha^{(R)} \otimes I_n\big) \\ I_M \otimes \big(\alpha \odot_{\mathrm{KR}} \alpha\big) \end{bmatrix}, \tag{6}$$

where $\otimes$ denotes the Kronecker product, and $\odot_{\mathrm{KR}}$ denotes the Khatri–Rao (column-wise Kronecker) product:

$$\alpha \odot_{\mathrm{KR}} \alpha = \left[\alpha^{(1)} \otimes \alpha^{(1)}, \ldots, \alpha^{(R)} \otimes \alpha^{(R)}\right] \in \mathbb{R}^{n^2\times R}.$$

Each of the first R block rows in (6) has dimension $(n^2M)\times n$, and the last block row $I_M \otimes \big(\alpha \odot_{\mathrm{KR}} \alpha\big)$ has dimension $(n^2M)\times(MR)$, so that $\partial\,\mathrm{vec}(\Theta)/\partial\gamma \in \mathbb{R}^{(n^2M)\times(n+M)R}$. See **Appendix** A for a detailed derivation of $\partial\,\mathrm{vec}(\Theta)/\partial\gamma$.

Combining (4)–(6) yields the full expression for $\partial\mathcal{P}_{i,j,\cdot}/\partial\gamma$, and the complete formulation is presented in **Appendix** B.

2.5. Covariance structure and estimation

The cross-layer dependencies within each node pair (i,j) are summarized by the covariance matrices $\Sigma_{i,j} = \mathrm{Cov}(\mathcal{A}_{i,j,\cdot})$. In line with the GEE framework, we approximate these true covariances by a parsimonious working covariance family that assumes a common correlation structure across node pairs:

$$\Sigma^{\mathrm{w}}_{i,j}(\gamma) = \Gamma_{i,j}^{1/2} W \Gamma_{i,j}^{1/2}, \tag{7}$$

where $\Gamma_{i,j} \in \mathbb{R}^{M\times M}$ is a diagonal matrix whose (m,m)-th entry is $\mathcal{P}_{i,j,m}(\gamma)\big(1-\mathcal{P}_{i,j,m}(\gamma)\big)$, and $W \in \mathbb{R}^{M\times M}$ is a positive-definite correlation matrix shared across node pairs. The working covariance matrices appearing in (3) are then

$$\widehat{\Sigma}_{i,j} = \Gamma_{i,j}^{1/2} \widehat{W} \Gamma_{i,j}^{1/2},$$

where $\widehat{W}$ is an empirical estimate of W.

We estimate W by pooling residuals across all node pairs:

$$\widehat{W} = \tfrac{1}{N}\textstyle\sum_{i\le j} \Gamma_{i,j}^{-1/2}\big(\mathcal{A}_{i,j,\cdot} - \mathcal{P}_{i,j,\cdot}(\hat{\gamma})\big)\big(\mathcal{A}_{i,j,\cdot} - \mathcal{P}_{i,j,\cdot}(\hat{\gamma})\big)^\top \Gamma_{i,j}^{-1/2} \tag{8}$$

where $\hat{\gamma}$ is the current estimate of γ, $N = n(n+1)/2$ is the number of node pairs with $i \le j$, and the sum runs over all such pairs, consistent with our convention in Section 2.3. Under mild regularity conditions, the eigenvalues of $\Sigma^{\mathrm{w}}_{i,j}(\gamma)$ are bounded away from zero and infinity, which ensures numerical stability and underpins our asymptotic analysis in Section 3. In practice, to avoid numerical instability when inverting $\widehat{\Sigma}_{i,j}$, we may add a small ridge term ϵI_M to $\widehat{W}$.

2.6. Optimization and computational complexity

In practice, we do not solve the nonlinear estimating equations (3) in closed form. Instead, we optimize γ using iterative gradient-based updates, alternating with periodic updates of the working correlation matrix W.

Optimization procedure. At a high level, each training epoch consists of: (i) computing the CP-based parameter tensor Θ and the corresponding edge probabilities $\mathcal{P}_{i,j,m}(\gamma)$ for sampled node pairs (i, j) and layers m; (ii) evaluating the score contributions in (3) via the Jacobian $\partial\mathcal{P}_{i,j,.}/\partial\gamma$; and (iii) updating γ by a gradient step (with optional regularization), followed by an update of W using (8). In our implementation we initialize W as the identity I_M (an independence working structure) and start updating it after a few warm-up epochs.

Per-iteration complexity and scalability. In the dense case, computing the CP-based parameter tensor Θ and the corresponding edge probabilities $\mathcal{P}_{i,j,m}(\gamma)$ for all node pairs and layers requires $O(Rn^2M)$ operations per full pass, similar to standard CP factorization. Evaluating the score function in (3) has the same order, since it aggregates contributions over all (i, j, m).

In most real-world settings, multilayer networks are sparse. Using sparse tensor representations and mini-batching over observed edges, the dominant cost per iteration scales as $O(R|E|)$, where $|E|$ is the total number of observed edges across all layers. The update of the working correlation W in (8) is computed from aggregated residuals and can be performed infrequently (e.g., every K epochs), so its amortized overhead is small compared to the main embedding updates. These properties make T-GINEE applicable to moderate-scale multilayer graphs (e.g., n in the thousands and M in the tens) when combined with sparse tensor implementations. Scaling to large-scale graphs necessitates efficient sampling strategies; a detailed discussion on this adaptation is provided in **Appendix** C.

3. Theoretical Results of T-GINEE

In this section, we delve into the foundational theoretical properties of the T-GINEE method. To rigorously establish its performance, we begin by outlining a set of essential assumptions that define the statistical framework within which T-GINEE operates. These assumptions are critical for deriving the consistency and asymptotic normality of the estimator, which are subsequently presented and discussed in detail. A more complete framework and full proofs are provided in **Appendix** D.

3.1. Assumptions

Assumption 3.1 (Boundedness)**.** All involved random variables and their derivatives are uniformly bounded. There exist constants $0 < \varepsilon < 1/2$ and $C < \infty$ such that for all (i, j, m) and for all parameters γ in a neighborhood of γ_0,

$$\boxed{|\mathcal{A}_{i,j,m}| \le C, \qquad \varepsilon \le \mathcal{P}_{i,j,m}(\gamma) \le 1-\varepsilon, \qquad |g'(\mathcal{P}_{i,j,m}(\gamma))| \le C} \tag{9}$$

The derivatives of g up to the required order are uniformly bounded in a neighborhood of γ_0. For the sparsity-aware logit links described in Section 2.3, the upper bound $1-\varepsilon$ can be replaced by $s-\varepsilon$, where $s \in (0, 1)$ is the sparsity coefficient.

Assumption 3.1 ensures that all relevant random variables and their derivatives remain within controlled limits, preventing extreme values that could destabilize the estimation process. By bounding these quantities, we can effectively manage the behavior of the estimator and its derivatives in the vicinity of the true parameter γ_0.

Assumption 3.2 (Identifiability)**.** The true parameter tensor Θ_0 admits a rank-R CP decomposition that is identifiable up to permutation and scaling of factors. We assume that the effective parameter dimension $(n+M)R$ grows sufficiently slowly with n so that the Fisher information and the relevant Hessians remain well-conditioned. A concrete sufficient condition used in Theorem 3.6 is $(n + M)R = o(n^{1/3})$, which is satisfied in our empirical settings where R is a few dozen and n is several hundred.

Assumption 3.2 guarantees that the true parameter tensor Θ_0 can be uniquely decomposed into its constituent factors, up to permutation and scaling. This identifiability is crucial for accurately recovering the underlying model parameters from the data. Additionally, by constraining the growth rate of $(n + M)R$ relative to n, this assumption ensures that the Hessian matrices remain invertible, which is necessary for the consistency and asymptotic normality of the estimator. The identifiability condition in Assumption 3.2 is standard in CP factorization: it can be ensured, for example, under Kruskal-type conditions on the factor matrices $\alpha_0 \in \mathbb{R}^{n\times R}$ and $\beta_0 \in \mathbb{R}^{M\times R}$. In particular, if the Kruskal ranks k_α and k_β satisfy $k_\alpha + k_\alpha + k_\beta \ge 2R + 2$, then the rank-$R$ CP decomposition of Θ_0 is unique up to permutation and scaling of the components, which justifies treating $\gamma_0 = [\mathrm{vec}(\alpha_0)^\top, \mathrm{vec}(\beta_0)^\top]^\top$ as a well-defined target parameter.

Assumption 3.3 (Working Covariance)**.** The true covariance matrices $\Sigma_{i,j}$ are positive definite with eigenvalues bounded away from zero and infinity. The working covariance $\widehat{\Sigma}_{i,j}$ satisfies

$$\left\|\widehat{\Sigma}_{i,j}^{-1} - \widetilde{\Sigma}_{i,j}^{-1}\right\|_F = O_p(N^{-1/2})$$

for some positive definite $\widetilde{\Sigma}_{i,j}$ with bounded eigenvalues. Correlation misspecification is allowed, as long as it converges at this rate.

Assumption 3.3 pertains to the properties of the covariance matrices used in the model. By requiring the true covariance matrices $\Sigma_{i,j}$ to be positive definite with eigenvalues bounded away from zero and infinity, we ensure numerical stability and prevent issues related to ill-conditioning.

Furthermore, allowing for correlation misspecification that converges at a rate of $O_p(N^{-1/2})$ provides flexibility while maintaining the validity of asymptotic results. Under the independent-pair assumption, $\widehat{W}$ is the average of N (approximately) i.i.d. matrix-valued terms with finite second moments and thus converges to its population counterpart at the usual parametric rate $O_p\big(N^{-1/2}\big)$ (Van der Vaart, 2000).

Assumption 3.4 (Moment Conditions)**.** The residuals $\big(\mathcal{A}_{i,j,\cdot} - \mathcal{P}_{i,j,\cdot}(\Theta_0)\big)$, scaled by their standard deviations, have sub-Gaussian tails. There exists $\delta > 0$ such that

$$\max_{i,j} \mathbb{E}\Big[\big\|\Sigma_{i,j}^{-1/2}\big(\mathcal{A}_{i,j,\cdot} - \mathcal{P}_{i,j,\cdot}(\Theta_0)\big)\big\|^{2+\delta}\Big] < \infty.$$

This ensures suitable Lindeberg-type (Ash & Doléans-Dade, 2000; Van der Vaart, 2000; Brown, 1971) conditions for central limit arguments.

Assumption 3.4 imposes specific moment conditions on the residuals of the model. This assumption facilitates applications of central limit theorem-type arguments by ensuring that the scaled residuals have sub-Gaussian tails and possess finite $(2+\delta)$ moments. These conditions are essential for establishing the asymptotic normality of the estimator, as they control the influence of extreme residuals and guarantee the convergence of the estimator's distribution.

Assumption 3.5 (Smoothness)**.** There exists a true parameter tensor $\Theta_0 \in \mathbb{R}^{n\times n\times M}$ with rank R that admits a unique CP decomposition $\Theta_0 = \mathcal{I} \times_1 \alpha_0 \times_2 \alpha_0 \times_3 \beta_0$. The link function g is three-times continuously differentiable with uniformly bounded first, second, and third derivatives. The partial derivatives of $\mathcal{P}(\gamma)$ with respect to γ are bounded, and the Hessian matrices with respect to (α, β) are well-conditioned in a neighborhood of $\gamma_0 = [\mathrm{vec}(\alpha_0)^\top, \mathrm{vec}(\beta_0)^\top]^\top$.

Assumption 3.5 addresses the smoothness and differentiability of both the link function g and the parameter tensor Θ_0. The requirement that g is three-times continuously differentiable with bounded derivatives allows for the use of Taylor expansions and other analytical techniques in the proofs of consistency and asymptotic normality. Additionally, ensuring that the partial derivatives of $\mathcal{P}(\gamma)$ are bounded and that the Hessian matrices are well-conditioned supports the stability and reliability of the parameter estimates.

3.2. Main Theoretical Results

For convenience, define the score function $s(\gamma)$ as

$$s(\gamma) = \sum_{i\le j} \left(\frac{\partial \mathcal{P}_{i,j,\cdot}}{\partial \gamma}\right)^\top \widehat{\Sigma}_{i,j}^{-1}\big(\mathcal{A}_{i,j,\cdot} - \mathcal{P}_{i,j,\cdot}(\gamma)\big). \quad (10)$$

With the necessary lemmas and assumptions in place, we now establish the main theoretical guarantees of T-GINEE. We first show consistency and then prove normality.

Theorem 3.1 (Consistency)**.** *Under Assumptions 3.1–3.5, There exists a solution $\hat{\gamma}$ to $s(\gamma) = 0$ such that*

$$\|\hat{\gamma} - \gamma_0\|_2 = O_p(N^{-1/2}),$$

where $N = n(n+1)/2$, $\|\cdot\|_2$ denotes Euclidean norm.

Proof. A detailed proof is provided in Appendix D.2. □

Theorem 3.2 (Asymptotic normality)**.** *Under Assumptions 3.1–3.5, we have*

$$\sqrt{N}\,(\hat{\gamma} - \gamma_0) \xrightarrow{d} \mathcal{N}(0, \Omega),$$

where $N = n(n+1)/2$ is the number of node pairs (i, j) with $i \le j$, $\Omega = M(\gamma_0)^{-1}B(\gamma_0)M(\gamma_0)^{-1}$, $M(\gamma_0)$ is the limit of $N^{-1}\nabla s(\gamma_0)$, and $B(\gamma_0)$ is the limit of $\mathrm{Var}(N^{-1/2}s(\gamma_0))$.

Proof. A detailed proof is provided in Appendix D.3. □

Corollary 3.3. *Under Assumptions 3.1–3.5, replacing $\Sigma_{i,j}^{-1}$ by $\widehat{\Sigma}_{i,j}^{-1}$ in the score function $s(\gamma)$ alters its value at γ_0 by only an $O_p(\sqrt{N})$ term. Formally, if $\widetilde{s}(\gamma)$ is defined in the same way as $s(\gamma)$ but uses $\widetilde{\Sigma}_{i,j}^{-1}$ instead of $\widehat{\Sigma}_{i,j}^{-1}$, then*

$$\|s(\gamma_0) - \widetilde{s}(\gamma_0)\| = O_p(\sqrt{N}).$$

Proof. A detailed proof is provided in Appendix D.4. □

The above theorems provide foundational theoretical guarantees for the T-GINEE method. The consistency result (Theorem 3.1) ensures that the estimator $\hat{\gamma}$ converges to the true parameter γ_0 as the number of node pairs $N = n(n+1)/2$ increases, with the estimation error decreasing at a rate of $N^{-1/2}$ (i.e., of order $1/n$ since $N \asymp n^2$). This establishes the reliability of T-GINEE in accurately estimating the underlying parameters in large samples.

In addition, Corollary 3.3 shows that replacing the true covariance $\Sigma_{i,j}^{-1}$ with an estimated version $\widehat{\Sigma}_{i,j}^{-1}$ in the score function still yields only an $o_p(\sqrt{N})$ difference at γ_0. This indicates that minor covariance misspecifications do not materially affect the key asymptotic properties of T-GINEE.

Furthermore, the asymptotic normality result (Theorem 3.2) characterizes the distribution of the estimator $\hat{\gamma}$, demonstrating that after appropriate scaling, it converges to a multivariate normal distribution. Due to space limitations, we provide additional remarks and discussion in **Appendix** E.

4. Experiments

In this section, we conduct comprehensive experiments to evaluate our T-GINEE framework.

Table 1. Link prediction performance (AUC) on synthetic multilayer network.

Method	CP	Tucker	NMF	SVD	LSE	MASE	NNTUCK	SPECK	HOSVD	T-GINEE
AUC	0.4488	0.5291	0.7216	0.8130	0.2234	0.3821	0.6105	0.7603	0.8503	**0.9395**

4.1. Experiment Settings

To comprehensively evaluate the performance of our proposed **T-GINEE** model, we conduct experiments on four benchmark multilayer network datasets, each capturing distinct real-world relational structures. Due to space limitations, detailed descriptions of datasets, baselines, and implementation details are provided in **Appendix** F.

4.2. Synthetic Data Results

To evaluate our method in a controlled environment, we generate synthetic multilayer networks with known correlation structures using a parameterized model:

$$\mathcal{P}_{i,j,m} = \rho \cdot \mathcal{P}_{i,j}^{\text{base}} + (1-\rho) \cdot U_{i,j,m}, \qquad \mathcal{A}_{i,j,m} = \mathbf{1}\{V_{i,j,m} < \mathcal{P}_{i,j,m}\}, \tag{11}$$

Here, $\rho = 0.2$ controls inter-layer correlation, $\mathcal{P}_{i,j}^{\text{base}}$ is a shared base probability matrix, and $U_{i,j,m}$ and $V_{i,j,m}$ are i.i.d. $\text{Unif}[0,1]$ noise variables. We construct networks with $n = 100$ nodes and $M = 3$ layers for link prediction tasks.

As shown in Table 1, T-GINEE achieves the highest AUC score of 0.9395, substantially outperforming all baselines including the second-best HOSVD. The significant performance gap between tensor-based methods and simpler approaches like LSE (0.2234) and MASE (0.3821) confirms the importance of explicitly modeling multilayer dependencies. Furthermore, the dramatic improvement of T-GINEE over basic CP decomposition (0.4488) demonstrates the effectiveness of our statistical regularization framework in capturing complex inter-layer correlations, validating our theoretical analysis.

4.3. Real-World Results

Based on the experimental results shown in Table 2, our proposed T-GINEE model demonstrates superior performance across datasets of varying scales and complexities compared to baseline methods. On the standard benchmark datasets (AUCS, Krackhardt, WAT, and Yeast), T-GINEE achieves the highest AUC scores of 0.920, 0.948, 0.838, and 0.921 respectively, outperforming all baseline methods in our experiments. Among the baseline methods, traditional matrix factorization approaches, such as SVD and NMF, show relatively strong performance, with SVD achieving the second-best results on AUCS (0.877) and Krackhardt (0.932). HOSVD, as a tensor-based method, also demonstrates competitive performance, particularly on the AUCS and Yeast datasets. However, simpler methods such as CP decomposition and LSE exhibit limited effectiveness, with LSE performing poorly on Yeast.

Table 2. Performance comparison of different methods. "oom" denotes Out-Of-Memory errors.

Method	AUC score on different datasets					
	AUCS	Krackhardt	WAT	Yeast	dblp	stackoverflow
CP	0.374	0.354	0.454	0.397	oom	oom
Tucker	0.487	0.702	0.580	<u>0.745</u>	oom	oom
NMF	<u>0.848</u>	<u>0.921</u>	0.707	0.863	0.6505	0.9642
SVD	0.877	0.932	<u>0.719</u>	0.879	<u>0.6093</u>	<u>0.9682</u>
LSE	0.297	0.384	0.153	0.047	0.6302	oom
MASE	0.480	0.361	0.342	0.347	oom	oom
NNTUCK	0.500	0.521	0.741	0.667	oom	oom
SPECK	0.793	0.658	0.655	0.903	oom	oom
HOSVD	<u>0.897</u>	0.783	0.820	0.902	oom	oom
T-GINEE	**0.920**	**0.948**	**0.838**	**0.921**	0.6478	**0.9831**

To comprehensively evaluate the scalability of T-GINEE, we extended our experiments to two large-scale real-world networks: DBLP, an academic collaboration network with up to 300,000 nodes, and Stack Overflow, a massive temporal interaction network with approximately 2.6 million nodes. As indicated in the rightmost columns of Table 2, most traditional tensor-based methods, including CP, Pure-Tucker, HOSVD, and MASE, failed to process these large-scale datasets, resulting in Out-Of-Memory (OOM) errors. This highlights the severe computational bottleneck of standard tensor decompositions when applied to million-node graphs. In contrast, T-GINEE successfully scaled to these massive datasets while maintaining superior accuracy. While matrix-based methods like SVD and NMF could handle the scale, T-GINEE surpassed them in performance. On the DBLP dataset, T-GINEE achieved an AUC of 0.6478, outperforming SVD (0.6093). On the massive Stack Overflow dataset, T-GINEE maintained high accuracy (0.9831), exceeding the best matrix baselines.

The significant performance gap between T-GINEE and other methods can be attributed to its ability to capture high-order structural patterns and incorporate both local and global network information. The consistent superior performance across diverse datasets, ranging from small biological networks to massive social platforms, validates the model's generalizability and stability. Notably, the performance improvement is particularly pronounced on the WAT dataset, where T-GINEE outperforms the second-best method by a margin of 0.018 in AUC, suggesting that our model is especially effective in handling complex network structures. For further evidence of T-GINEE's effectiveness, see the triangle prediction analysis in **Appendix** G.

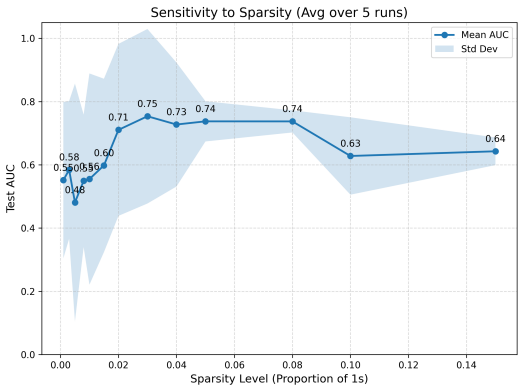


Figure 2. Sensitivity of T-GINEE to graph sparsity on synthetic multilayer networks. We vary the proportion of observed edges (proportion of 1 entries) and report the mean test AUC over 5 runs with one standard deviation as the shaded region.

4.4. Sensitivity to Sparsity

We further study how T-GINEE performs as multilayer networks vary in sparsity. Using the synthetic setup in Section 4.2, we adjust the Bernoulli generator so the proportion of observed edges (i.e., 1 entries in the adjacency tensor) ranges from below 1% to about 15%. At each sparsity level, we train T-GINEE with the hyperparameters from Section 4.2 and run 5 trials with different seeds. Figure 2 reports mean test AUC with one standard deviation. AUC increases from the extremely sparse regime to moderate sparsity, peaking at roughly AUC ≈ 0.75, then remains fairly stable and declines only gradually for dense graphs, staying above AUC ≈ 0.63 even at 10–15% edge density. Overall, T-GINEE is robust across a broad sparsity range: performance is weaker and more variable when edges are extremely sparse, but stabilizes once a moderate amount of edge information is available.

Due to space constraints, a detailed investigation into the impact of embedding dimension and regularization weight on model performance and computational efficiency is presented in **Appendix** H.

5. Related Work

Network embedding. Network embedding learns low-dimensional node representations that preserve network structure. Early methods include matrix factorization techniques such as SVD (Golub & Reinsch, 1970) and NMF (Lee & Seung, 2001; Cai et al., 2011). Random walk-based approaches, including DeepWalk (Perozzi et al., 2014) and node2vec (Grover & Leskovec, 2016), adapt word embedding techniques to networks. More recently, graph convolutional networks (GCNs) (Hamilton et al., 2017) have become popular for incorporating node features and modeling complex relations. However, these methods are ill-suited for multilayer systems because they either treat layers independently or use simplistic aggregation, losing inter-layer dependencies (Wang et al., 2017b; Dong et al., 2017). Such aggregation, often simple summation or concatenation of layer-specific embeddings, can obscure the distinct and complementary roles of different relationship types. T-GINEE leverages generalized estimating equations to explicitly capture cross-layer dependencies.

Multilayer graph analysis and embedding. Multilayer graphs offer rich representations for complex systems (Kivelä et al., 2014; De Domenico et al., 2013; Boccaletti et al., 2014). While tensor methods like CP decomposition are natural extensions (Wang et al., 2017a), a recent survey (Yousefzadeh et al., 2025) notes three limitations in current embedding methods: (i) neglecting cross-layer dependencies by treating layers independently (Papalexakis et al., 2013; Boden et al., 2017); (ii) assuming unrealistic complete node correspondence; and (iii) lacking theoretical guarantees in deep models like M-DeepWalk (Song & Thiagarajan, 2018), node2vec variants (Liu et al., 2017), MGCN (Ghorbani et al., 2019), and MR-GCN (Huang et al., 2020). T-GINEE addresses the first and third of these issuesby integrating CP decomposition with generalized estimating equations (GEE) (Liang & Zeger, 1986), while, like most existing work, still assuming a shared node set across layers. Unlike prior matrix factorization(Tang et al., 2009; Gligorijević et al., 2016), random walk (Song & Thiagarajan, 2018; Liu et al., 2017), or GCN methods (Ghorbani et al., 2019; Huang et al., 2020), our framework explicitly models inter-layer correlations with rigorous statistical guarantees. Thus, T-GINEE offers a grounded backbone complementary to expressive deep encoders.

6. Conclusion

We propose T-GINEE, a tensor-based generalized estimating equation framework for multilayer graph representation learning that explicitly models cross-network dependencies through a principled statistical formulation. By combining CP decomposition with GEE, T-GINEE makes a central theoretical contribution: establishing the consistency and asymptotic normality of embeddings under mild regularity conditions, thereby providing rigorous statistical guarantees. Experiments demonstrate its effectiveness on synthetic and real-world networks, highlighting the robust mathematical foundation T-GINEE offers for analyzing complex interdependent systems. Limitations include potential constraints on extremely sparse or large-scale networks and a priority on statistical validation over engineering optimizations like attention-based architectures. Future work will explore combining our objective with deep encoders and extending the framework to dynamic settings. Details about limitations and LLM usage are provided in **Appendix** I and J.

Impact Statement

This paper presents work whose goal is to advance the field of machine learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

References

Aguiar, I., Taylor, D., and Ugander, J. A tensor factorization model of multilayer network interdependence, 2024. URL https://arxiv.org/abs/2206.01804.

Alves, L. G., Mangioni, G., Cingolani, I., Rodrigues, F. A., Panzarasa, P., and Moreno, Y. The nested structural organization of the worldwide trade multi-layer network. *Scientific reports*, 9(1):2866, 2019.

Ash, R. B. and Doléans-Dade, C. A. *Probability and measure theory*. Academic press, 2000.

Backstrom, L., Huttenlocher, D., Kleinberg, J., and Lan, X. Group formation in large social networks: membership, growth, and evolution. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 44–54, 2006.

Boccaletti, S., Bianconi, G., Criado, R., Del Genio, C. I., Gómez-Gardenes, J., Romance, M., Sendiña-Nadal, I., Wang, Z., and Zanin, M. The structure and dynamics of multilayer networks. *Physics Reports*, 544(1):1–122, 2014.

Boden, B., Günnemann, S., Hoffmann, H., and Seidl, T. Mimag: mining coherent subgraphs in multi-layer graphs with edge labels. *Knowledge and Information Systems*, 50(2):417–446, 2017.

Brown, B. M. Martingale central limit theorems. *The Annals of Mathematical Statistics*, pp. 59–66, 1971.

Cai, D., He, X., Han, J., and Huang, T. S. Graph regularized nonnegative matrix factorization for data representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1548–1560, 2011.

Cai, H., Zheng, V. W., and Chang, K. C.-C. A comprehensive survey of graph embedding: Problems, techniques, and applications. *IEEE transactions on knowledge and data engineering*, 30(9):1616–1637, 2018.

De Domenico, M., Solé-Ribalta, A., Cozzo, E., Kivelä, M., Moreno, Y., Porter, M. A., Gómez, S., and Arenas, A. Mathematical formulation of multilayer networks. *Physical Review X*, 3(4):041022, 2013.

De Domenico, M., Nicosia, V., Arenas, A., and Latora, V. Structural reducibility of multilayer networks. *Nature Communications*, 6:6864, 04 2015. doi: 10.1038/ncomms7864.

Dong, X., Frossard, P., Vandergheynst, P., and Nefedov, N. Clustering with multi-layer graphs: A spectral perspective. *IEEE Transactions on Signal Processing*, 60(11): 5820–5831, 2012.

Dong, Y., Chawla, N. V., and Swami, A. metapath2vec: Scalable representation learning for heterogeneous networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 135–144, 2017.

Ghorbani, M., Baghshah, M. S., and Rabiee, H. R. Mgcn: semi-supervised classification in multi-layer graphs with graph convolutional networks. In *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pp. 208–211, 2019.

Gligorijević, V., Panagakis, Y., and Zafeiriou, S. Fusion and community detection in multi-layer graphs. In *2016 23rd international conference on pattern recognition (ICPR)*, pp. 1327–1332. IEEE, 2016.

Golub, G. H. and Reinsch, C. Singular value decomposition and least squares solutions. *Numerische mathematik*, 14 (5):403–420, 1970.

Greene, D. and Cunningham, P. Producing a unified graph representation from multiple social network views. In *Proceedings of the 5th annual ACM web science conference*, pp. 118–121, 2013.

Grover, A. and Leskovec, J. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 855–864, 2016.

Hamilton, W., Ying, Z., and Leskovec, J. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30, 2017.

Han, Q., Xu, K., and Airoldi, E. Consistent estimation of dynamic and multi-layer block models. In *International Conference on Machine Learning*, pp. 1511–1520. PMLR, 2015.

Huang, Z., Li, X., Ye, Y., and Ng, M. K. Mr-gcn: Multi-relational graph convolutional networks based on generalized tensor product. In *IJCAI*, volume 20, pp. 1258–1264, 2020.

Interdonato, R., Magnani, M., Perna, D., Tagarelli, A., and Vega, D. Multilayer network simplification: approaches,

models and methods. *Computer Science Review*, 36: 100246, 2020.

Jiao, P., Lu, R., Jin, D., Wang, Y., and Wu, H. An effective and robust framework by modeling correlations of multiplex network embedding. In *2021 IEEE International Conference on Data Mining (ICDM)*, pp. 1144–1149. IEEE, 2021.

Jing, B.-Y., Li, T., Lyu, Z., and Xia, D. Community detection on mixture multilayer networks via regularized tensor decomposition. *The Annals of Statistics*, 49(6): 3181–3205, 2021.

Kipf, T. N. and Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Kivelä, M., Arenas, A., Barthelemy, M., Gleeson, J. P., Moreno, Y., and Porter, M. A. Multilayer networks. *Journal of Complex Networks*, 2(3):203–271, 2014.

Kolda, T. G. and Bader, B. W. Tensor decompositions and applications. *SIAM review*, 51(3):455–500, 2009.

Krackhardt, D. Cognitive social structures. *Social Networks*, 9(2):109–134, 1987. ISSN 0378-8733. doi: https://doi.org/10.1016/0378-8733(87)90009-8.

Lee, D. D. and Seung, H. S. Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems*, pp. 556–562, 2001.

Lei, J., Chen, K., and Lynch, B. Consistent community detection in multi-layer network data. *Biometrika*, 107 (1):61–73, 2020.

Li, D., Pan, Z., Hu, G., Anderson, G., and He, S. Active module identification from multilayer weighted gene co-expression networks: a continuous optimization approach. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 18(6):2239–2248, 2020.

Liang, K.-Y. and Zeger, S. L. Longitudinal data analysis using generalized linear models. *Biometrika*, 73(1):13–22, 1986.

Liu, W., Chen, P.-Y., Yeung, S., Suzumura, T., and Chen, L. Principled multilayer network embedding. In *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*, pp. 134–141. IEEE, 2017.

Liu, X., Maiorino, E., Halu, A., Glass, K., Prasad, R. B., Loscalzo, J., Gao, J., and Sharma, A. Robustness and lethality in multilayer biological molecular networks. *Nature communications*, 11(1):6043, 2020.

Liu, X., Guan, W., Li, L., Li, H., Lin, C., Li, X., Chen, S., Xu, J., Deng, H., and Zheng, B. Pretraining representations of multi-modal multi-query e-commerce search. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 3429–3437, 2022.

Lyu, Z., Xia, D., and Zhang, Y. Latent space model for higher-order networks and generalized tensor decomposition. *Journal of Computational and Graphical Statistics*, 32(4):1320–1336, 2023.

Paatero, P. and Tapper, U. Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 5(2):111–126, 1994.

Papalexakis, E. E., Akoglu, L., and Ience, D. Do more views of a graph help? community detection and clustering in multi-graphs. In *Proceedings of the 16th International Conference on Information Fusion*, pp. 899–905. IEEE, 2013.

Paranjape, A., Benson, A. R., and Leskovec, J. Motifs in temporal networks. In *Proceedings of the tenth ACM international conference on web search and data mining*, pp. 601–610, 2017.

Paul, S. and Chen, Y. Spectral and matrix factorization methods for consistent community detection in multi-layer networks. *The Annals of Statistics*, 48(1):230–250, 2020. doi: 10.1214/18-AOS1800.

Pereyra, V. and Scherer, G. Efficient computer manipulation of tensor products with applications to multidimensional approximation. *Math. Comp.*, 27:595–605, 1973.

Perozzi, B., Al-Rfou, R., and Skiena, S. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 701–710, 2014.

Ren, Z.-M., Zeng, A., and Zhang, Y.-C. Bridging nestedness and economic complexity in multilayer world trade networks. *Humanities and Social Sciences Communications*, 7(1):1–8, 2020.

Rossi, L. and Magnani, M. Towards effective visual analytics on multiplex and multilayer networks. *Chaos, Solitons & Fractals*, 72:68–76, 2015. ISSN 0960-0779. doi: https://doi.org/10.1016/j.chaos.2014.12.022. Multiplex Networks: Structure, Dynamics and Applications.

Shanthamallu, U. S., Thiagarajan, J. J., Song, H., and Spanias, A. Gramme: Semisupervised learning using multilayered graph attention models. *IEEE transactions on neural networks and learning systems*, 31(10):3977–3988, 2019.

Song, H. and Thiagarajan, J. J. Improved deep embeddings for inferencing with multi-layered networks. *arXiv preprint arXiv:1811.12156*, 2018.

Tang, W., Lu, Z., and Dhillon, I. S. Clustering with multiple graphs. In *2009 Ninth IEEE International Conference on Data Mining*, pp. 1016–1021. IEEE, 2009.

Tucker, L. R. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31(3):279–311, 1966. doi: 10.1007/BF02289464.

Van Den Oord, E. J. and Van Rossem, R. Differences in first graders' school adjustment: The role of classroom characteristics and social structure of the group. *Journal of School Psychology*, 40(5):371–394, 2002.

Van der Vaart, A. W. *Asymptotic statistics*, volume 3. Cambridge university press, 2000.

Wang, D., Wang, H., and Zou, X. Identifying key nodes in multilayer networks based on tensor decomposition. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 27(6), 2017a.

Wang, X., Cui, P., Wang, J., Pei, J., Zhu, W., and Yang, S. Community preserving network embedding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017b.

Xing, E., Jordan, M., Russell, S. J., and Ng, A. Distance metric learning with application to clustering with side-information. *Advances in neural information processing systems*, 15, 2002.

Xu, F., Li, Y., and Xu, S. Attentional multi-graph convolutional network for regional economy prediction with open migration data. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2225–2233, 2020.

Xu, K., Hu, W., Leskovec, J., and Jegelka, S. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.

Yang, C., Pal, A., Zhai, A., Pancha, N., Han, J., Rosenberg, C., and Leskovec, J. Multisage: Empowering gcn with contextualized multi-embeddings on web-scale multipartite networks. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 2434–2443, 2020.

Yeung, K. Y., Medvedovic, M., and Bumgarner, R. E. Clustering gene-expression data with repeated measurements. *Genome Biology*, 4(5):R34, 2003. doi: 10.1186/gb-2003-4-5-r34.

Yousefzadeh, N., Thai, M. T., and Ranka, S. A comprehensive survey on multi-layer graph embedding methods. *Vietnam Journal of Computer Science*, pp. 1–40, 2025.

Yuan, Y. and Qu, A. Community detection with dependent connectivity. *The Annals of Statistics*, 49(4):2378–2428, 2021.

Zhang, H., Qiu, L., Yi, L., and Song, Y. Scalable multiplex network embedding. In *IJCAI*, volume 18, pp. 3082–3088, 2018.

Zhang, X., Li, L., Zhou, H., Zhou, Y., Shen, D., et al. Tensor generalized estimating equations for longitudinal imaging analysis. *Statistica Sinica*, 29(4):1977, 2019.

Zheng, W., Wang, D., and Zou, X. Control of multilayer biological networks and applied to target identification of complex diseases. *BMC bioinformatics*, 20:1–12, 2019.

A. Proof of Derivations $\frac{\partial \operatorname{vec}(\Theta)}{\partial \gamma}$

We first consider a rank-one tensor $\mathcal{T} = a \circ a \circ c$ with $a \in \mathbb{R}^n$ and $c \in \mathbb{R}^M$. By the definition of $\operatorname{vec}(\mathcal{T})$, we have

$$\operatorname{vec}(\mathcal{T}) = \big(c_1(a \otimes a)^\top, \ldots, c_M(a \otimes a)^\top\big)^\top.$$

Denote $\{e_i\}_{i=1}^n$ as the canonical basis in $\mathbb{R}^n$. For one thing, note that

$$\begin{aligned}\frac{\partial(a \otimes a)}{\partial a} &= \begin{bmatrix} a^\top + a_1 e_1^\top & a_2 e_1^\top & \ldots & a_n e_1^\top \\ a_1 e_2^\top & a^\top + a_2 e_2^\top & \ldots & a_n e_2^\top \\ \vdots & \vdots & \ddots & \vdots \\ a_1 e_n^\top & a_2 e_n^\top & \ldots & a^\top + a_n e_n^\top \end{bmatrix}^\top \\ &= I_n \otimes a + a \otimes I_n, \end{aligned} \tag{12}$$

which leads to

$$\frac{\partial \operatorname{vec}(\mathcal{T})}{\partial a} = c^\top \otimes \big(I_n \otimes a + a \otimes I_n\big). \tag{13}$$

For another, it is clear that

$$\frac{\partial \operatorname{vec}(\mathcal{T})}{\partial c} = I_M \otimes (a \otimes a). \tag{14}$$

According to the CP decomposition of Θ, we have

$$\operatorname{vec}(\Theta) = \sum_{r=1}^R \big(\beta_1^{(r)}(\alpha^{(r)} \otimes \alpha^{(r)})^\top, \ldots, \beta_M^{(r)}(\alpha^{(r)} \otimes \alpha^{(r)})^\top\big)^\top.$$

By the property (13), we have

$$\frac{\partial \operatorname{vec}(\Theta)}{\partial \operatorname{vec}(\alpha)} = \begin{bmatrix} (\beta^{(1)})^\top \otimes \big(I_n \otimes \alpha^{(1)} + \alpha^{(1)} \otimes I_n\big) \\ (\beta^{(2)})^\top \otimes \big(I_n \otimes \alpha^{(2)} + \alpha^{(2)} \otimes I_n\big) \\ \vdots \\ (\beta^{(R)})^\top \otimes \big(I_n \otimes \alpha^{(R)} + \alpha^{(R)} \otimes I_n\big) \end{bmatrix}. \tag{15}$$

By the property (14), we have

$$\frac{\partial \operatorname{vec}(\Theta)}{\partial \operatorname{vec}(\beta)} = \begin{bmatrix} I_M \otimes \big(\alpha^{(1)} \otimes \alpha^{(1)}\big) \\ I_M \otimes \big(\alpha^{(2)} \otimes \alpha^{(2)}\big) \\ \vdots \\ I_M \otimes \big(\alpha^{(R)} \otimes \alpha^{(R)}\big) \end{bmatrix} = I_M \otimes \big(\alpha \odot_{\mathrm{KR}} \alpha\big), \tag{16}$$

The desired result follows from (15) and (16) immediately.

B. Full Formulation and Approximation of T-GINEE

Putting all pieces together, the T-GINEE in (3) is approximated by

$$\begin{aligned}\sum_{i \le j} \begin{bmatrix} (\beta^{(1)})^\top \otimes \Delta^{(1)} \\ \vdots \\ (\beta^{(R)})^\top \otimes \Delta^{(R)} \\ I_M \otimes \big(\alpha \odot_{\mathrm{KR}} \alpha\big) \end{bmatrix} \begin{bmatrix} \operatorname{vec}\big(\mathcal{E}^{(i,j,1)}\big)^\top \\ \vdots \\ \operatorname{vec}\big(\mathcal{E}^{(i,j,M)}\big)^\top \end{bmatrix}^\top \\ \times \operatorname{diag}\big(g'(\mathcal{P}_{i,j,1}), \ldots, g'(\mathcal{P}_{i,j,M})\big)^{-1} \widehat{\Sigma}_{i,j}^{-1}\big(\mathcal{A}_{i,j,\cdot} - \mathcal{P}_{i,j,\cdot}(\gamma)\big) = \mathbf{0}, \end{aligned} \tag{17}$$

where $\Delta^{(r)} = I_n \otimes (\alpha^{(r)})^\top + (\alpha^{(r)})^\top \otimes I_n$ for $r \in [R]$ and $\widehat{\Sigma}_{i,j} = \Gamma_{i,j}^{1/2} \widehat{W} \Gamma_{i,j}^{1/2}$ is the estimated covariance matrix.

C. Scalability to Massive Graphs

Scaling tensor-based methods to massive graphs (e.g., $n > 10^6$ nodes) presents significant challenges. Standard tensor decompositions typically operate on the full adjacency tensor $\mathcal{A} \in \mathbb{R}^{n \times n \times M}$, where n is the number of nodes. For massive datasets like Stack Overflow ($n \approx 2.6 \times 10^6$), instantiating this tensor is infeasible. Instead, we adopt an edge-centric perspective. Let $\mathcal{E}^+$ denote the set of observed positive edges across all layers. We define a sparse dataset $\mathcal{D} = \{(u, v, m) \mid \mathcal{A}_{u,v,m} = 1\}$.

Dynamic Negative Sampling. To avoid the computationally prohibitive cost of iterating over all non-existent edges (zeros in the tensor), we employ dynamic negative sampling during training. For each mini-batch of positive samples $\mathcal{B}^+ \subset \mathcal{E}^+$, we generate a corresponding set of negative samples $\mathcal{B}^-$. For a positive triplet $(u, v, m) \in \mathcal{B}^+$, we sample a negative node $v' \in \mathcal{V}$ uniformly at random to construct (u, v', m). We assume the collision probability (where (u, v') is actually a positive edge) is negligible for sparse massive graphs. This strategy reduces the per-epoch computational complexity from $O(n^2 \cdot M)$ to $O(|\mathcal{E}| \cdot M)$.

C.1. Batch-wise GEE with Momentum Update

The core of T-GINEE is the Generalized Estimating Equation (GEE) framework, which traditionally requires estimating the working covariance matrix $\mathbf{W}$ using residuals from all node pairs. To adapt this to mini-batch training, we propose a **Batch-GEE Loss** combined with a momentum update mechanism.

Batch-GEE Loss Function. We reformulate the global GEE objective into a differentiable loss function calculated over a mini-batch $\mathcal{B} = \mathcal{B}^+ \cup \mathcal{B}^-$. The total loss $\mathcal{L}_{total}$ is defined as:

$$\mathcal{L}_{total} = \mathcal{L}_{BCE} + \lambda \cdot \mathcal{L}_{GEE}(\mathcal{B}; \mathbf{W}), \tag{18}$$

where $\mathcal{L}_{BCE}$ is the standard binary cross-entropy loss measuring reconstruction error. The regularization term $\mathcal{L}_{GEE}$ captures inter-layer correlations and is defined as:

$$\mathcal{L}_{GEE}(\mathcal{B}; \mathbf{W}) = \frac{1}{|\mathcal{B}|} \sum_{(i,j) \in \mathcal{B}} \mathbf{r}_{ij}^T \mathbf{\Sigma}_{ij}^{-1} \mathbf{r}_{ij}, \tag{19}$$

where $\mathbf{r}_{ij} \in \mathbb{R}^M$ is the residual vector for node pair (i, j) across M layers, calculated as $\mathbf{r}_{ij} = \mathbf{a}_{ij} - \mathbf{p}_{ij}(\gamma)$. Here, $\mathbf{a}_{ij}$ is the observed edge vector and $\mathbf{p}_{ij}$ is the predicted probability vector. The matrix $\mathbf{\Sigma}_{ij}^{-1}$ is the inverse of the working covariance for pair (i, j), approximated via the global correlation structure $\mathbf{W}$. Specifically, we utilize the standardized residuals $\tilde{\mathbf{r}}_{ij} = \mathbf{\Gamma}_{ij}^{-1/2} \mathbf{r}_{ij}$, where $\mathbf{\Gamma}_{ij}$ is the diagonal variance matrix derived from the predicted probabilities. The GEE term simplifies to the quadratic form $\tilde{\mathbf{r}}_{ij}^T \mathbf{W}^{-1} \tilde{\mathbf{r}}_{ij}$.

Momentum Update for Working Covariance. A critical challenge in batch training is that the global correlation structure $\mathbf{W}$ cannot be accurately estimated from a single batch. To resolve this, we maintain a global buffer for $\mathbf{W}$ and update it using a momentum-based moving average. In iteration t, let $\mathbf{W}_{batch}^{(t)}$ be the empirical correlation matrix computed from the current batch's standardized residuals. The global $\mathbf{W}$ is updated as:

$$\mathbf{W}^{(t)} \leftarrow \alpha \mathbf{W}^{(t-1)} + (1 - \alpha) \mathbf{W}_{batch}^{(t)}, \tag{20}$$

where $\alpha \in [0, 1)$ is the momentum coefficient (set to 0.9 in our experiments). This approach ensures that $\mathbf{W}$ stabilizes to represent the global cross-layer dependency structure while allowing efficient mini-batch optimization.

C.2. Complexity Analysis

The Scalable T-GINEE framework significantly reduces resource requirements. Regarding **Space Complexity**, by storing only the model parameters (embeddings $\alpha \in \mathbb{R}^{n \times R}$, $\beta \in \mathbb{R}^{M \times R}$) and the edge list, the space complexity is $O((n + M)R + |\mathcal{D}|)$, which is linear with respect to the number of nodes and observed edges, avoiding the $O(n^2)$ bottleneck. As for **Time Complexity**, the cost per training iteration is proportional to the batch size $|\mathcal{B}|$. The total time complexity per epoch is $O(|\mathcal{D}| \cdot R)$, i.e., linear in the number of observed edge triplets and the embedding rank, making it feasible to train on datasets with millions of nodes (e.g., Stack Overflow) using standard GPU or even CPU hardware.

D. Derivations of Theorems

D.1. Lemmas

Lemma D.1. *Let $N = n(n+1)/2$ denote the number of node pairs (i, j) with $i \leq j$. Under Assumptions 3.1–3.5, consider the initial estimator obtained by solving*

$$\sum_{i \leq j} \left(\frac{\partial \mathcal{P}_{i,j,\cdot}}{\partial \gamma} \right)^\top (\mathcal{A}_{i,j,\cdot} - \mathcal{P}_{i,j,\cdot}(\gamma)) = 0,$$

using an independence working structure ($\Sigma_{i,j} = I_M$). Then, the initial estimator $\tilde{\gamma}$ is $O_p(N^{-1/2})$-consistent for the true parameter γ_0.

Proof. Consider the estimating equation defined by

$$s(\gamma) = \sum_{i \leq j} \left(\frac{\partial \mathcal{P}_{i,j,\cdot}}{\partial \gamma} \right)^\top (\mathcal{A}_{i,j,\cdot} - \mathcal{P}_{i,j,\cdot}(\gamma)) \,.$$

Under Assumption 3.1, all random variables involved, including $\mathcal{A}_{i,j,m}$, $\mathcal{P}_{i,j,m}(\gamma)$, and the derivatives of the link function g, are uniformly bounded for all indices (i, j, m) and for all parameter vectors γ in a neighborhood of the true parameter γ_0. Assumption 3.5 ensures that the partial derivatives $\frac{\partial \mathcal{P}_{i,j,\cdot}(\gamma)}{\partial \gamma}$ are bounded and that the link function g is thrice continuously differentiable with uniformly bounded first and second derivatives.

By the law of large numbers, as n (and hence $N = n(n+1)/2$) tends to infinity, the normalized sum $N^{-1}s(\gamma)$ converges in probability to its expectation. Specifically, at the true parameter value γ_0, the expectation of each term in the sum satisfies

$$\mathbb{E}\left[\left(\frac{\partial \mathcal{P}_{i,j,\cdot}}{\partial \gamma} \right)^\top (\mathcal{A}_{i,j,\cdot} - \mathcal{P}_{i,j,\cdot}(\gamma_0)) \right] = 0,$$

since $\mathbb{E}[\mathcal{A}_{i,j,\cdot}] = \mathcal{P}_{i,j,\cdot}(\gamma_0)$ by the model specification.

Assumption 3.2 guarantees that the true parameter γ_0 is uniquely identifiable as the solution to $\mathbb{E}\{s(\gamma)\} = 0$. Moreover, the low-rank condition $(n + M)R$ growing sufficiently slowly relative to n ensures that the dimensionality does not impede the identification or the invertibility of the relevant Hessian matrix.

Define the normalized Jacobian

$$D_N(\gamma) := -N^{-1} \frac{\partial s(\gamma)}{\partial \gamma}.$$

Expanding $N^{-1}s(\gamma)$ around γ_0 using a Taylor series, we obtain

$$N^{-1}s(\tilde{\gamma}) = N^{-1}s(\gamma_0) + D_N(\gamma_0)(\tilde{\gamma} - \gamma_0) + r_N,$$

where $\|r_N\| = o_p(N^{-1/2})$ by Assumption 3.5. From Lemma D.2, $D_N(\gamma_0)$ is invertible with eigenvalues bounded away from zero and infinity, ensuring that $D_N(\gamma_0)$ is well-conditioned.

Solving the linear approximation for $\tilde{\gamma}$ yields

$$\tilde{\gamma} - \gamma_0 = -D_N(\gamma_0)^{-1}N^{-1}s(\gamma_0) + o_p(N^{-1/2}).$$

The term $N^{-1/2}s(\gamma_0)$ is a sum of N mean-zero random vectors with uniformly bounded moments; under Assumption 3.4 and by the central limit theorem,

$$N^{-1/2}s(\gamma_0) = O_p(1).$$

Since $D_N(\gamma_0)$ is non-singular and its inverse has bounded operator norm, it follows that

$$\tilde{\gamma} - \gamma_0 = O_p(N^{-1/2}).$$

Therefore, the initial estimator $\tilde{\gamma}$ converges to the true parameter γ_0 at the rate $N^{-1/2}$ in probability, establishing its $O_p(N^{-1/2})$-consistency. □

Lemma D.2. *Let $N = n(n+1)/2$ and define the normalized Jacobian*

$$D_N(\gamma) = -N^{-1}\frac{\partial s(\gamma)}{\partial \gamma}.$$

Under Assumptions 3.1–3.5, the matrix $D_N(\gamma_0)$ is invertible with eigenvalues bounded away from zero and infinity. Furthermore,

$$\sup_{\|\gamma-\gamma_0\|\le 4N^{-1/2}} \big\|D_N(\gamma) - D_N(\gamma_0)\big\| = O_p\big(N^{-1/2}\big).$$

Proof. By definition,

$$\begin{aligned} D_N(\gamma) &= -N^{-1}\frac{\partial s(\gamma)}{\partial \gamma} \\ &= N^{-1}\sum_{i\le j}\left[\left(\frac{\partial^2 \mathcal{P}_{i,j,\cdot}(\gamma)}{\partial\gamma\partial\gamma^\top}\right)(\mathcal{A}_{i,j,\cdot} - \mathcal{P}_{i,j,\cdot}(\gamma)) + \left(\frac{\partial \mathcal{P}_{i,j,\cdot}(\gamma)}{\partial\gamma}\right)\left(\frac{\partial \mathcal{P}_{i,j,\cdot}(\gamma)}{\partial\gamma}\right)^\top\right]. \end{aligned}$$

Under Assumption 3.5, the second derivatives $\frac{\partial^2 \mathcal{P}_{i,j,\cdot}(\gamma)}{\partial\gamma\partial\gamma^\top}$ are uniformly bounded for γ in a neighborhood of γ_0, and the first derivatives are also uniformly bounded. This ensures that each summand in $D_N(\gamma)$ is $O(1)$ in operator norm.

Assumption 3.2 implies that at $\gamma = \gamma_0$,

$$D_N(\gamma_0) = N^{-1}\sum_{i\le j}\left(\frac{\partial \mathcal{P}_{i,j,\cdot}(\gamma_0)}{\partial\gamma}\right)\left(\frac{\partial \mathcal{P}_{i,j,\cdot}(\gamma_0)}{\partial\gamma}\right)^\top$$

converges to a positive-definite limit with eigenvalues bounded away from zero and infinity. Hence $D_N(\gamma_0)$ is invertible and well-conditioned.

To analyze $D_N(\gamma) - D_N(\gamma_0)$, observe that for $\|\gamma-\gamma_0\| \le 4N^{-1/2}$, the smoothness of $\mathcal{P}(\gamma)$ implies that

$$\left\|\frac{\partial \mathcal{P}_{i,j,\cdot}(\gamma)}{\partial\gamma} - \frac{\partial \mathcal{P}_{i,j,\cdot}(\gamma_0)}{\partial\gamma}\right\| \le L\|\gamma-\gamma_0\| \le 4LN^{-1/2},$$

for some Lipschitz constant L that does not depend on (i,j) or N. A similar bound holds for the second derivatives.

Consequently, each summand in $D_N(\gamma) - D_N(\gamma_0)$ differs by at most $O(N^{-1/2})$ in operator norm. Taking the average over N node pairs yields

$$\big\|D_N(\gamma) - D_N(\gamma_0)\big\| \le N^{-1}\sum_{i\le j} O(N^{-1/2}) = O(N^{-1/2}),$$

uniformly over $\|\gamma-\gamma_0\| \le 4N^{-1/2}$. This establishes the desired bound and the lemma follows. □

D.2. Proof of Theorem 3.1

Proof. Let $N = n(n+1)/2$. From Lemma D.1, we know that the initial estimator $\tilde{\gamma}$ satisfies

$$\|\tilde{\gamma} - \gamma_0\| = O_p(N^{-1/2}).$$

Now consider the full estimator $\hat{\gamma}$ obtained by solving the estimating equation $s(\gamma) = 0$. Expanding the normalized score $N^{-1}s(\hat{\gamma})$ around γ_0 via a Taylor series, we get

$$0 = N^{-1}s(\hat{\gamma}) = N^{-1}s(\gamma_0) + D_N(\gamma_0)(\hat{\gamma}-\gamma_0) + r_N,$$

where $D_N(\gamma) = -N^{-1}\partial s(\gamma)/\partial\gamma$ as in Lemma D.2, and $\|r_N\| = o_p(N^{-1/2})$ due to the smoothness and boundedness conditions in Assumption 3.5.

By Lemma D.2, $D_N(\gamma_0)$ is invertible with eigenvalues bounded away from zero and infinity. Solving for $\hat{\gamma}-\gamma_0$ gives

$$\hat{\gamma} - \gamma_0 = -D_N(\gamma_0)^{-1}N^{-1}s(\gamma_0) - D_N(\gamma_0)^{-1}r_N.$$

Under Assumption 3.4 and the independence of node pairs, $N^{-1/2}s(\gamma_0)$ is a sum of N mean-zero random vectors with uniformly bounded moments, so

$$N^{-1/2}s(\gamma_0) = O_p(1).$$

Combined with the boundedness of $D_N(\gamma_0)^{-1}$ and the fact that $\|r_N\| = o_p(N^{-1/2})$, this implies

$$\|\hat{\gamma} - \gamma_0\| = O_p(N^{-1/2}),$$

establishing the consistency of the estimator. □

D.3. Proof of Theorem 3.2

Proof. Let $N = n(n+1)/2$. The proof follows from the central limit theorem and the consistency result established in Theorem 3.1. First, we establish a central limit result for $s(\gamma_0)$. By Assumption 3.4 and the independence of node pairs (i, j),

$$\frac{s(\gamma_0)}{\sqrt{N}} \xrightarrow{d} \mathcal{N}\big(0, B(\gamma_0)\big),$$

where $B(\gamma_0)$ is the limit of $\mathrm{Var}\big(N^{-1/2}s(\gamma_0)\big)$.

From the consistency proof in Theorem 3.1, we know that

$$0 = N^{-1}s(\hat{\gamma}) = N^{-1}s(\gamma_0) + D_N(\gamma_0)(\hat{\gamma} - \gamma_0) + o_p(N^{-1/2}).$$

Rearranging this expression gives

$$\sqrt{N}(\hat{\gamma} - \gamma_0) = -D_N(\gamma_0)^{-1}\frac{s(\gamma_0)}{\sqrt{N}} + o_p(1).$$

By Lemma D.2 and Assumption 3.2, $D_N(\gamma_0)$ converges to a non-singular limit $M(\gamma_0)$ with eigenvalues bounded away from zero and infinity. Applying the continuous mapping theorem leads to the asymptotic distribution

$$\sqrt{N}(\hat{\gamma} - \gamma_0) \xrightarrow{d} \mathcal{N}\Big(0,\ M(\gamma_0)^{-1}B(\gamma_0)\big[M(\gamma_0)^{-1}\big]^\top\Big).$$

Thus, the estimator $\hat{\gamma}$ is asymptotically normal with mean γ_0 and covariance matrix $\Omega = M(\gamma_0)^{-1}B(\gamma_0)\big[M(\gamma_0)^{-1}\big]^\top$, which matches the form stated in Theorem 3.2. □

D.4. Covariance Estimation Corollary

Corollary. *Let $N = n(n+1)/2$. Under Assumptions 3.1–3.5, replacing $\Sigma_{i,j}^{-1}$ by $\widehat{\Sigma}_{i,j}^{-1}$ in the score function $s(\gamma)$ alters its value at γ_0 by only an $O_p(\sqrt{N})$ term. Formally, if $\widetilde{s}(\gamma)$ is defined in the same way as $s(\gamma)$ but uses $\widetilde{\Sigma}_{i,j}^{-1}$ instead of $\widehat{\Sigma}_{i,j}^{-1}$, then*

$$\|s(\gamma_0) - \widetilde{s}(\gamma_0)\| = O_p(\sqrt{N}).$$

Proof. By Assumption 3.3, we have

$$\big\|\widehat{\Sigma}_{i,j}^{-1} - \widetilde{\Sigma}_{i,j}^{-1}\big\|_F = O_p\big(N^{-1/2}\big).$$

Since all remaining factors in the construction of $s(\gamma)$ are uniformly bounded (Assumption 3.1) and satisfy appropriate moment conditions (Assumption 3.4), the difference introduced by $\widehat{\Sigma}_{i,j}^{-1}$ versus $\widetilde{\Sigma}_{i,j}^{-1}$ contributes at most $O_p(N^{-1/2})$ to each term in $s(\gamma_0)$. Summing over all (i, j) (there are N node pairs) yields

$$N \times O_p\big(N^{-1/2}\big) = O_p\big(\sqrt{N}\big),$$

which is negligible at the $\sqrt{N}$-scale relevant for the asymptotic distribution of $\hat{\gamma}$. Hence $\|s(\gamma_0) - \widetilde{s}(\gamma_0)\| = O_p(\sqrt{N})$, as claimed. □

Corollary D.4 shows that using a slightly misspecified or estimated covariance in the score function $s(\gamma)$ does not affect the key asymptotic rate at γ_0. This result is crucial in ensuring that minor estimation errors in the covariance structure remain inconsequential for the consistency and asymptotic distribution of the parameter estimates. In practice, it allows us to work with convenient or empirically estimated covariance matrices without compromising the main theoretical guarantees.

E. A Few Remarks on T-GINEE

First, T-GINEE is related to, but different from, generalized estimating equations (GEE, (Liang & Zeger, 1986)) or tensor generalized estimating equations (TGEE, (Zhang et al., 2019)) for generalized multivariate linear regression models with correlated predictors. Clearly, the multi-layer network $\mathcal{A}$ plays the role of the response variable in GEE or TGEE. However, there is no edgewise covariate to be regressed on, and the mean of $\mathcal{A}$ contains nothing but the parameters to be estimated.

Second, the rationale behind T-GINEE is that seeking a solution is a relaxation to minimize the following quadratic form

$$\frac{1}{2}\sum_{i\le j}\big(\mathcal{A}_{i,j,\cdot}-\mathcal{P}_{i,j,\cdot}(\Theta)\big)^{\top}\Sigma_{i,j}^{-1}\big(\mathcal{A}_{i,j,\cdot}-\mathcal{P}_{i,j,\cdot}(\Theta)\big). \tag{21}$$

This is because the left-hand side of (3) is essentially the negative gradient of (21). Herein, the precision matrix $\Sigma_{i,j}^{-1}$ serves as the metric matrix (Xing et al., 2002; Liu et al., 2022) to measure the deviation of $\mathcal{A}_{i,j,\cdot}$ to its expectation $\mathcal{P}_{i,j,\cdot}$. When the edges $\mathcal{A}_{i,j,m}$ for $m\in[M]$ are independent, we have $\Sigma_{i,j}=I_M$, the M-dimensional identity matrix, and (21) reduces to the least squares loss

$$\frac{1}{4}\big\|\mathcal{A}-\mathcal{P}(\Theta)\big\|_F^2-\frac{1}{4}\sum_{i=1}^{n}\big\|\mathcal{A}_{i,i,\cdot}-\mathcal{P}_{i,i,\cdot}(\Theta)\big\|^2.$$

The framework of least squares estimation for network data has been popularly employed in the literature (Paul & Chen, 2020; Lei et al., 2020).

Third, a trivial solution to the GINEE (3) as well as the minimizer of the quadratic form (21) is $\mathcal{P}=\mathcal{A}$ if there is no further constraint in $\mathcal{P}$ or Θ. This solution is meaningless and has no implication for downstream tasks of network analysis, such as network embedding, community detection, node classification, change point detection, and sub-graph density estimation. Moreover, the numbers of samples (the $\mathcal{A}_{i,j,m}$ with $i\le j$), free parameters in Θ, and unique equations in (3) are all $n(n+1)M/2$ due to the semi-symmetry of the multi-layer network. Thus, it is necessary to reduce the number of free parameters in Θ in order to derive a consistent estimator for Θ or $\mathcal{P}$ for subsequent tasks of multi-layer network analysis.

Fourth, selecting an appropriate rank R is a crucial practical issue for tensor-based models such as T-GINEE, since it directly affects both accuracy and efficiency. Our experiments suggest a clear trade-off: higher ranks yield better accuracy but demand more computational resources. For applications where predictive performance is paramount (for example, biological network analysis), moderately high ranks ($R=32$–64) are recommended, while in resource-constrained or real-time settings, smaller ranks ($R=8$–16) provide balanced accuracy and efficiency. A practical guideline is to set

$$R\approx C\log\big(\min\{n,M\}\big),$$

where n is the number of nodes and M is the number of layers, and C is a modest constant calibrated on validation data.

F. Datasets, Baselines and Implementation Details

F.1. Datasets

- **Krackhardt** (Krackhardt, 1987): This dataset records the cognitive social structures of a management team in a high-tech manufacturing firm, consisting of 21 managers. Each manager reported their perceived advice relationships with others, resulting in a $21\times21\times21$ tensor, where each layer corresponds to an individual's perception of the advice network.
- **AUCS** (Rossi & Magnani, 2015): The AUCS dataset consists of 61 individuals in a university setting, with five types of pairwise relations: current working, leisure activities, lunch companionship, co-authorship, and Facebook friendship. These networks form a $61\times61\times5$ multilayer adjacency tensor, facilitating the study of social group structures and community detection.
- **YSCGC** (Yeung et al., 2003): This gene co-expression dataset contains 205 genes under four functional categories, observed over 4 replicated experimental conditions. The multilayer network is constructed by thresholding pairwise gene expression similarities, resulting in a $205\times205\times4$ binary adjacency tensor for community detection and functional module discovery.
- **WAT** (De Domenico et al., 2015): The World Agricultural Trade (WAT) dataset describes trading relationships of 130 major countries across 32 agricultural products in 2010. We represent this as a $130\times130\times32$ multilayer network, with each layer indicating trade interactions for a specific product. This dataset is used for multilayer link prediction tasks.

- **Stack Overflow** (Paranjape et al., 2017): The Stack Overflow dataset captures temporal interactions between users on the technical Q&A platform. We construct a temporal multiplex network by discretizing continuous timestamps into 5 chronological snapshots. We represent this as a $2.58 \times 10^6 \times 2.58 \times 10^6 \times 5$ tensor, containing approximately 48 million interactions. This dataset serves as a large-scale benchmark to evaluate the scalability and performance of the model on massive temporal graphs.

- **DBLP** (Backstrom et al., 2006): The DBLP dataset describes the co-authorship network of computer science researchers. We represent this as an $N \times N \times 5$ multilayer network, where the five layers correspond to distinct temporal snapshots of co-authorship history. Each layer indicates whether two researchers co-authored a paper during that specific time period. This dataset is used to evaluate the model's scalability and link prediction performance on varying graph sizes up to $N \approx 300,000$.

These datasets encompass diverse network sizes and structural properties, ranging from standard benchmarks to large-scale networks such as DBLP (up to 300k nodes) and the massive Stack Overflow dataset (over 2.5 million nodes), providing a robust testbed for the effectiveness and generalizability of T-GINEE in multilayer network representation learning, community detection, and link prediction.

F.2. Evaluation Metrics

Model performance was primarily evaluated using the Area Under the ROC Curve (AUC) on the test set, a standard metric for link prediction tasks. Additionally, we tracked both the binary cross-entropy (BCE) loss component (measuring prediction accuracy) and the GEE loss component (measuring correlation structure modeling) throughout training.

F.3. Implementation Details

The T-GINEE model was implemented in PyTorch, leveraging CP decomposition for efficient tensor factorization of multilayer graphs. The architecture employs node embeddings $\alpha \in \mathbb{R}^{n \times d}$ and layer embeddings $\beta \in \mathbb{R}^{M \times d}$, constructing a parameter tensor $\Theta \in \mathbb{R}^{n \times n \times M}$ through CP decomposition, which is passed through a logistic function to predict edge probabilities. After hyperparameter tuning, we selected an embedding dimension $d = 32$ for our synthetic dataset experiments, using the Adam optimizer with a learning rate of 0.01 and weight decay of 10^{-5}. Training proceeded with a batch size of 10,000 edges for 50 epochs, with the regularization weight between BCE loss and GEE loss set to 0.1. The working covariance matrix was updated every 5 epochs with a smoothing factor of 0.9.

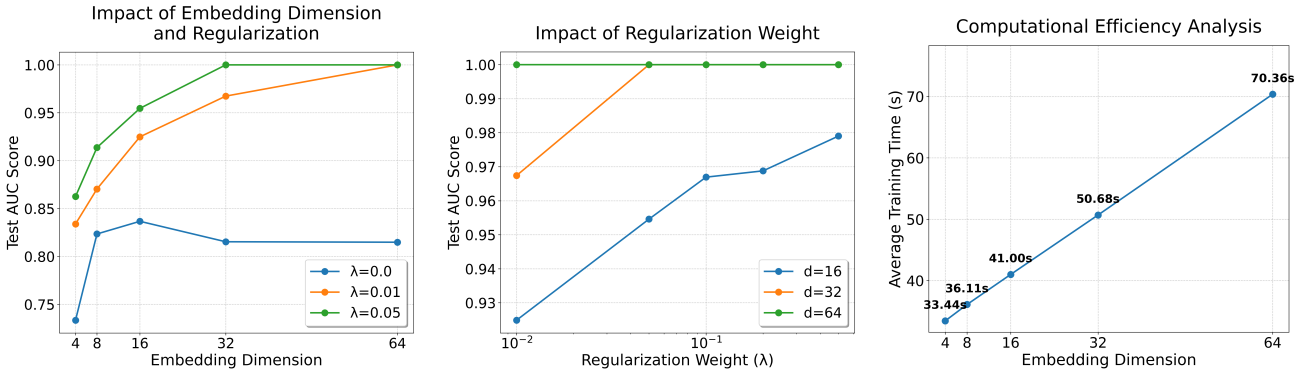
We performed a grid search over embedding dimensions $d \in \{16, 32\}$, learning rates in $\{0.001, 0.01\}$, and regularization weights in $\{0.01, 0.1, 0.5\}$, selecting the configuration with highest validation AUC. Dataset partitioning followed an $80\%/10\%/10\%$ split for training/validation/testing with a fixed random seed of 42. All experiments were conducted with PyTorch 2.2.2, with an average training time of approximately 20 minutes per dataset.

F.4. Baselines

To comprehensively evaluate the effectiveness of our proposed T-GINEE model, we compare it against a diverse set of baseline methods, encompassing classical spectral algorithms, tensor decompositions, and matrix factorization approaches. The *Mean Adjacency Spectral Embedding (MASE)* (Han et al., 2015) approach computes the average adjacency matrix across layers and performs spectral embedding via SVD, providing a simple yet effective baseline. The *Non-negative Tucker Decomposition (NNTuck)* (Aguiar et al., 2024) method performs a non-negative Tucker tensor factorization of the multilayer adjacency tensor, optimizing factor matrices using multiplicative updates under KL-divergence loss, with three variants considering different assumptions on layer interaction. The *Spectral Kernel-based Clustering (SPECK)* (Paul & Chen, 2020) aggregates spectral information from the Laplacian matrices of all layers, constructing a consensus embedding for clustering. *HOSVD-Tucker* (Jing et al., 2021) applies higher-order singular value decomposition with Tucker decomposition to the adjacency tensor, capturing intricate multiway interactions. *Layer-wise Spectral Embedding (LSE)* (Lei et al., 2020) performs spectral clustering on each layer independently before combining results through consensus or aggregation. *CP decomposition* (Pereyra & Scherer, 1973) factorizes the adjacency tensor into a sum of rank-one tensors using an alternating least-squares implementation, while *Tucker decomposition* (Tucker, 1966) generalizes CP by allowing a core tensor and separate factor matrices for each mode. *Non-negative Matrix Factorization (NMF)* (Paatero & Tapper, 1994) decomposes each adjacency matrix into non-negative factors, optionally weighting layers and applying ℓ_1 regularization, with consensus structure inferred via aggregation of factor matrices. Finally, *Singular Value Decomposition (SVD)* (Kolda & Bader, 2009) is

Table 3. Accuracy of triangular relationship prediction on the Krackhardt dataset.

Method	Accuracy
HOSVD	40.33%
SPECK	50.10%
NNTUCK	56.42%
T-GINEE	**73.36%**



(a) Impact of embedding dimension and regularization weight.

(b) Effect of regularization weight across different dimensions.

(c) Computational efficiency across dimensions.

Figure 3. Comprehensive analysis of model hyperparameters: (a) embedding dimension impact, (b) regularization effect, and (c) computational efficiency.

applied to each adjacency matrix or to the mean/concatenated adjacency to extract low-rank node embeddings.

G. Triangle Prediction on the Krackhardt Dataset

To further evaluate T-GINEE, we conducted a triangle prediction study on the Krackhardt dataset, which contains interpersonal relationship networks and is well-suited for cross-relationship prediction. We focused on triangular structures: for each triangle, one edge was removed during training and then predicted by the models. Table 3 reports accuracies across methods. T-GINEE achieves 73.36% accuracy, a 17% absolute improvement over the next best method (NNTUCK), thereby demonstrating its unique ability to leverage multilayer dependencies to infer missing relationships and validating the practical effectiveness of our theoretical framework.

H. Hyperparameter Analysis

We conduct a comprehensive hyperparameter analysis to investigate the impact of embedding dimension and regularization weight on model performance. All experiments are performed on the same dataset with consistent evaluation metrics.

Impact of embedding dimension. As shown in Figure 3a, the relationship between embedding dimension and model performance demonstrates clear patterns across different regularization settings. The experimental results show that increasing the embedding dimension generally improves model performance, with substantial gains observed when moving from 4 to 32 dimensions. Notably, with appropriate regularization ($\lambda = 0.05$), the model achieves perfect prediction accuracy (AUC $= 1.0$) when the embedding dimension reaches 32. Further increasing the dimension to 64 maintains this optimal predictive performance but introduces additional computational overhead.

Effect of regularization. The impact of the regularization weight is illustrated in Figure 3b. For larger dimensions (32 and 64), the model becomes more sensitive to regularization parameters, achieving optimal performance with smaller regularization weights ($\lambda = 0.01$–0.05). This suggests that proper regularization calibration is crucial for preventing overfitting in the embedding space, especially in higher-dimensional representation spaces where the model capacity increases substantially.

Computational efficiency. Figure 3c shows the relationship between embedding dimension and computational cost. The training time increases approximately linearly with the embedding dimension, from 33.22 seconds for 4-dimensional embeddings to 70.32 seconds for 64-dimensional embeddings. This linear scaling demonstrates the computational efficiency of our model, making it practical for real-world applications.

Based on the comprehensive results derived from these analyses, we recommend using an embedding dimension of 32 paired with a regularization weight of 0.05 as the default configuration, as it consistently provides optimal performance (AUC = 1.0) while maintaining reasonable computational efficiency. This configuration effectively strikes a robust balance between model expressiveness, generalization ability, and computational cost.

I. Limitations

While T-GINEE provides a robust framework for tensor-based multilayer graph representation learning, several limitations remain. First, the model relies on sufficient network density for accurate parameter estimation, which may constrain its effectiveness on extremely sparse large-scale networks. To mitigate this, we propose augmenting sparse graphs prior to embedding by employing graph completion techniques (e.g., link prediction) to infer missing edges. Additionally, our modified logit link function, $g(x) = \log(x/(s-x))$, incorporates a sparsity coefficient s that can adaptively accommodate varying density levels, allowing the model to maintain efficacy as sparsity increases.

From a theoretical perspective, our current asymptotic analysis relies on the condition that the effective parameter dimension $(n+M)R$ grows slower than $n^{1/3}$. In truly high-dimensional regimes where this condition is violated—such as scenarios with very large ranks R or extremely dense layer couplings—additional structural assumptions, such as sparsity constraints on the factors, may be required.

Finally, ethical and scope considerations must be addressed. While improved representation learning enhances predictive accuracy, its application to social networks warrants caution. Without appropriate privacy safeguards, granular modeling capabilities could potentially be misused for user profiling or surveillance. Furthermore, biases inherent in the input data may be preserved or amplified, necessitating rigorous fairness evaluations in sensitive applications. We also emphasize that the present study focuses on statistical regularization and does not currently incorporate graph neural networks or specific sparse-computation optimizations.

J. LLM Usage Disclosure

We disclose our use of large language models (LLMs) in preparing this manuscript. We employed OpenAI's ChatGPT and related tools for language-level support, including polishing writing, improving grammar, and enhancing clarity. The core scientific content of T-GINEE, including the theoretical development of tensor-based generalized estimating equations, formal proofs of consistency and asymptotic normality, and the design and execution of experiments on synthetic and real-world multilayer networks, was conceived, developed, and validated by the authors without LLM assistance. During manuscript preparation, LLMs were used selectively to (i) generate alternative formulations of technical explanations for readability, (ii) assist with retrieval and condensation of related work, and (iii) suggest phrasing for summarizing experimental results. In all cases, LLM outputs were reviewed, validated, and revised by the authors. For implementation, we did not rely on LLMs to design or optimize the core T-GINEE algorithm. LLMs were only used occasionally to refactor non-core utilities such as dataset preprocessing scripts, figure plotting functions, and LaTeX formatting of equations and tables. All quantitative results, model derivations, and inference procedures reported in this paper are the independent work of the authors and were verified without LLM involvement.

In summary, LLMs supported editing and presentation, while the substantive scientific contributions of T-GINEE are entirely original to the authors.