

---

# Graph-Theoretic Insights into Bayesian Personalized Ranking for Recommendation

---

Kai Zheng<sup>1,2,3</sup>, Jianxin Wang<sup>1,2,\*</sup>, Jinhui Xu<sup>4,5,\*</sup>

<sup>1</sup>School of Computer Science and Engineering, Central South University  
Changsha 410083, China

<sup>2</sup>The Hunan Provincial Key Lab of Bioinformatics, Central South University  
Changsha 410083, China

<sup>3</sup>The College of Computer Science and Technology, China University of Petroleum  
Qingdao 266580, China

<sup>4</sup>School of Information Science and Technology, University of Science and Technology of China  
HeFei, 230026, China

<sup>5</sup>Institute of Artificial Intelligence, Hefei Comprehensive National Science Center  
HeFei, 230026, China  
jxwang@mail.csu.edu.cn, jhxu@ustc.edu.cn

## Abstract

Graph self-supervised learning (GSL) is essential for processing graph-structured data, reducing the need for manual labeling. Traditionally, this paradigm has extensively utilized Bayesian Personalized Ranking (BPR) as its primary loss function. Despite its widespread application, the theoretical analysis of its node relations evaluation have remained largely unexplored. This paper employs recent advancements in latent hyperbolic geometry to deepen our understanding of node relationships from a graph-theoretical perspective. We analyze BPR’s limitations, particularly its reliance on local connectivity through 2-hop paths, which overlooks global connectivity and the broader topological structure. To address these shortcomings, we purpose a novel loss function, BPR+, designed to encompass even-hop paths and better capture global connectivity and topological nuances. This approach facilitates a more detailed measurement of user-item relationships and improves the granularity of relationship assessments. We validate BPR+ through extensive empirical testing across five real-world datasets and demonstrate its efficacy in refining graph self-supervised learning frameworks. Additionally, we explore the application of BPR+ in drug repositioning, highlighting its potential to support pharmaceutical research and development. Our findings not only illuminate the success factors of previous methodologies but also offer new theoretical insights into this learning paradigm.

## 1 Introduction

Self-supervised learning (SSL) is revolutionizing deep learning by significantly reducing the reliance on manually annotated labels [1]. This approach has gained considerable traction, especially in the application to graph-structured data, referred to as graph self-supervised learning (GSL) [2]. GSL exploits the unique, complex topological structures of nodes (non-Euclidean data space) to generate embeddings [2]. This contrasts sharply with the more straightforward Euclidean spaces typical of image and language data, presenting unique challenges and opportunities for innovation [1].

---

\*Corresponding authors

In GSL, designing an effective loss function capable of navigating the irregularities of graph data is crucial. Bayesian Personalized Ranking (BPR) has been widely adopted in GSL methods to address this need [3]. Originally formulated to solve the personalized ranking problem in recommendation systems, BPR assesses item rankings for a user based on their historical interactions [4]. The BPR loss function measures the relationships between users and items by calculating the dot product of their embeddings, differentiating connected pairs (positive samples) from unconnected pairs (negative samples) in the graph [5].

Despite BPR loss widespread use, the theoretical analysis of its relations evaluation within GSL remains a relatively unexplored area. In this paper, we analyze the physical underpinnings behind BPR loss from a graph theoretical perspective. Notably, based on the formula for BPR loss, the score  $y_{ui}$  can be interpreted as the computation of elements in the matrix  $\mathbf{E}\mathbf{E}^\top$ , specifically  $y_{ui} = (\mathbf{E}\mathbf{E}^\top)_{ui}$ . If we consider  $\mathbf{E}$  as an embedding network comprising users, items, and abstract nodes, then  $y_{ui}$  can be understood as a statistic of the 2-hop paths from user  $u$  to item  $i$  within this network. However, these 2-hop paths only capture local connectivity information. This coarse-grained assessment approach brings about two issues. Firstly, it lacks precision in measuring topological similarity, resulting in a single score corresponding to a wide range of topological similarities (e.g., from 20% to 50%). Secondly, these scores are influenced by the norm of embedding vectors, higher norm correspond to higher scores with other embeddings.

To address these limitations, it seems intuitive to extend our analysis to include longer n-hop paths. However, the selection of these paths and the quantification of their informational content pose formidable challenges. Previous study has shown that when the graph is sufficiently large, 2-hop paths approximate the concept of energy distance defined in latent hyperbolic geometry [6], derived using the principle of maximum entropy [7]. This measures the complement of the ratio of the latent distance between nodes in latent hyperbolic space to their chemical potential (a function of expected degree). Previous work indicates that as a measure of energy distance, 2-hop paths are not sufficiently precise. Consequently, we introduce the TopoLa distance to measure energy distance between nodes and, based on this, propose the BPR+ loss. We demonstrate that BPR+ loss more accurately measures the topological structure similarity and global connectivity information between users and items in the embedding network.

In addition to improve BPR loss, the introduction of network geometry (latent hyperbolic geometry) allows us to analyze common graph convolution operations from a new perspective and explain their effectiveness [6]. This not only clarifies the reasons behind the success of previous graph self-supervised learning but also provides a fresh perspective for the theoretical analysis of this learning paradigm. Extensive empirical tests of five existing methods on five real datasets confirm our theory and demonstrated the effectiveness of our new approach. Additionally, we apply BPR+ to the field of pharmaceutical research and development, developing a drug repositioning framework, TopoDR, and selecting ten potential therapeutic candidates for four prevalent cancers: colorectal, breast, stomach cancer, and leukemia.

**Contributions and paper structure.** We analyze Bayesian Personalized Ranking (BPR) loss from a graph-theoretical perspective, with a particular focus on the physical significance and its impact on embedding representations. In this process, we demonstrate that the measurement of relationships between users and items can be transformed into a path problem in graph theory, allowing us to use network geometry to aid the analysis. The main contributions are as follows:

- Demonstrate the limitations of BPR loss through latent hyperbolic geometry and provide relevant proofs (Section 3).
- Explore the effectiveness of common graph convolution operations in existing graph neural networks from the perspective of topology-encoded latent hyperbolic geometry topologically (Section 4).
- Propose a novel loss function, BPR+, as an alternative (Section 5).
- Develop an approximation technique based on numerical analysis to optimize the computation of loss. (Section 6).
- Apply BPR+ to the field of pharmaceutical research and development, introducing a novel computational framework for drug repositioning (Appendix F).

## 2 Preliminaries

**Bayesian Personalized Ranking.** BPR is a pairwise loss designed for one-class collaborative filtering [3]. The objective function of BPR encourages higher predictions for observed entries relative to their unobserved counterparts, formulated as follows:

$$L_{BPR} = - \sum_{u=1}^M \sum_{i \in \mathcal{N}_u} \sum_{j \notin \mathcal{N}_u} \ln \sigma(\hat{y}_{ui} - \hat{y}_{uj}) + \tau \left\| \mathbf{E}^{(0)} \right\|^2 \quad (1)$$

where  $\mathcal{N}_u$  represents the set of items associated with user  $u$ . The parameter  $\tau$  is utilized to regulate the intensity of L2 regularization.  $M$  denotes the number of users. The variable  $\hat{y}_{ui}$  is obtained through the product of the embeddings of user  $u$  and item  $i$ , that is  $\hat{y}_{ui} = \mathbf{e}_u \mathbf{e}_i^\top$ .

## 3 The limitation of BPR loss

In the Bayesian Personalized Ranking (BPR) loss, the score  $\hat{y}_{ui}$  between the user  $u$  and the item  $i$  is derived from the dot product of the embeddings of user  $u$  and item  $i$ . We find that computing  $\hat{y}_{ui}$  is equivalent to calculating the Gram matrix  $\mathbf{E}\mathbf{E}^\top$  of the embedding matrix  $\mathbf{E}$ . In graph theory, the Gram matrix  $\mathbf{E}\mathbf{E}^\top$  represents the number of 2-hop paths between user  $u$  and item  $i$ . During  $\hat{y}_{ui}$  computation, we model matrix  $\mathbf{E}$  as a graph comprising users, items, and  $N_e$  abstract nodes (Figure 1).

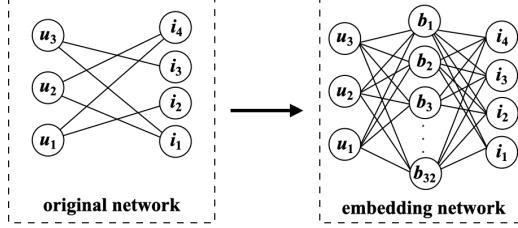


Figure 1: The illustration of embedding network.

Thus, graph self-supervised learning can be understood as the process of constructing an embedding network with a specified number of abstract nodes along with users and items to preserve the information of the original network. The weights of the edges in this network represent the relationships between the abstract nodes and other elements in the network. In BPR loss,  $\hat{y}_{ui}$  evaluates the interaction between user  $u$  and item  $i$  by assessing 2-hop paths via abstract nodes (Figure 2). Krioukov has demonstrated through the principle of maximum entropy that the counting of 2-hop paths positively correlates with the complement of the energy distance between nodes in the latent hyperbolic [7]. However, Previous research indicates that relying solely on this local connectivity information is insufficient for accurately measuring the energy distance between nodes in a latent hyperbolic space, as stated in **Theorem 1** (Appendix A). According to Theorem 1, we demonstrate that there exists a more precise method for measuring energy distance, which surpasses the method of counting 2-hop paths.

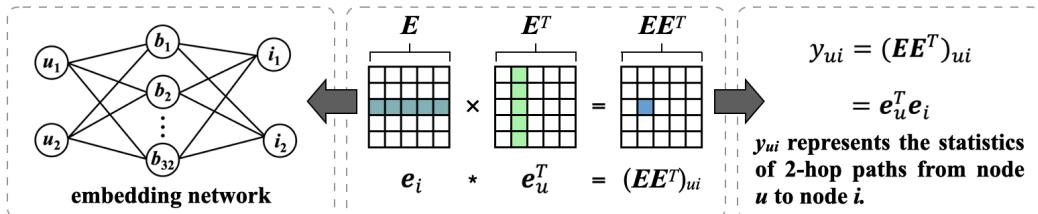


Figure 2: The physical significance of the Bayesian Personalized Ranking (BPR) loss.

**Theorem 1.** Given a network  $X \in \mathbb{R}^{(m+n) \times N_e}$ , let  $\langle t(x) \rangle$  denote the expected number of triangles, and  $\langle t'(x) \rangle$  represent the expected number of weighted triangles. Consequently,  $\alpha' > \alpha$ , where  $\alpha'$  is the logarithm of the thermodynamic activity corresponding to  $\langle t'(x) \rangle$ , and  $\alpha$  is associated with  $\langle t(x) \rangle$ .

## 4 Analyzing common graph convolution operations using network geometry

As shown in the previous section, the embedding matrix can be regarded as an embedding network. Consequently, we can utilize network geometry to analyze common graph convolution operations and explain their effectiveness. We use lightGCN as an example, a widely utilized foundational module.

**Light Graph Convolution (LGC).** Within the lightGCN, LGC is designed to focus on the fundamental elements of graph convolution, specifically information propagation and aggregation, while forgoing superfluous feature transformations and nonlinear activation functions. The operation streamlines the graph convolution process, enhancing the efficiency. The formulation of LGC is defined as follows:

$$e_u^{(k+1)} = \sum_{i \in \mathcal{N}_u} \frac{1}{\sqrt{|\mathcal{N}_u|} \sqrt{|\mathcal{N}_i|}} e_i^{(k)}, \quad e_i^{(k+1)} = \sum_{u \in \mathcal{N}_i} \frac{1}{\sqrt{|\mathcal{N}_i|} \sqrt{|\mathcal{N}_u|}} e_u^{(k)}, \quad (2)$$

where the symmetric normalization factor  $\frac{1}{\sqrt{|\mathcal{N}_r|} \sqrt{|\mathcal{N}_d|}}$ , in line with the conventional Graph Convolutional Network (GCN) architecture, is employed to curb the increase in embedding scale ensuing from successive graph convolution operations. The embedding of the  $k$ -th layer can be calculated as follows:

$$\mathbf{E}^{(k)} = \alpha_k \left( \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} \right)^k \mathbf{E}^{(0)}, \quad (3)$$

where  $\alpha_k \geq 0$  denotes the importance of the embeddings from the  $k$ -th layer in constituting the final embedding.  $\mathbf{D} \in \mathbb{R}^{(m+n) \times (m+n)}$  represents the degree matrix, where  $\mathbf{D}_{ii}$  indicates the degree of the  $i$ -th node in the adjacency matrix  $\mathbf{A}$ . From a graph theory perspective, the  $k$ -th layer of graph convolution symbolizes the single-step diffusion of the ID embedding  $\mathbf{E}^{(0)}$ , which is influenced by  $k$ -hop paths connecting user  $u$  and item  $i$  within  $\mathbf{A}$ . The symmetric normalization factor weights the statistics of  $k$ -hop paths to address their exponential increase.

**Layer Combination.** In the lightGCN architecture, the sole trainable elements are the ID embeddings:  $e_u^{(0)}$  and  $e_i^{(0)}$ . Post-initialization, embeddings obtained at  $k$ -th layer are calculated via LGC, as defined in Equation (3). Subsequently, the final network embedding are constructed through a weighted summation of embeddings from each layer:

$$\mathbf{e}_u = \sum_{k=0}^K \alpha_k \mathbf{e}_u^{(k)}, \quad \mathbf{e}_i = \sum_{k=0}^K \alpha_k \mathbf{e}_i^{(k)}, \quad (4)$$

where  $K$  is the number of layers. The matrix form of layer combination can be defined as:

$$\begin{aligned} \mathbf{E} &= \alpha_0 \mathbf{E}^{(0)} + \alpha_1 \mathbf{E}^{(1)} + \alpha_2 \mathbf{E}^{(2)} + \cdots \alpha_K \mathbf{E}^{(K)} \\ &= \alpha_0 \mathbf{E}^{(0)} + \alpha_1 \tilde{\mathbf{A}} \mathbf{E}^{(0)} + \alpha_2 \tilde{\mathbf{A}}^2 \mathbf{E}^{(0)} + \cdots \alpha_K \tilde{\mathbf{A}}^K \mathbf{E}^{(0)} \end{aligned} \quad (5)$$

where  $\tilde{\mathbf{A}} = \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}}$  is the symmetrically normalized matrix. From a graph theory perspective, there are three reasons why layer combination is effective: (1) With an increase in hops, the variance in the number of paths between nodes diminishes, leading to a reduction in the disparity between layer embeddings and explaining the tendency for over-smoothing at higher layers. (2) As demonstrated in previous work, path counts encapsulate information about node degrees and topological similarities between nodes. However, the extent to which these factors are reflected varies with the hop count, and fusing embeddings from multiple layers effectively captures this nuanced information. (3) This

operation has the same effect as self-connection in graph convolution. Essentially, self-connection adds a path connecting each node to itself.

## 5 The Improved Bayesian Personalized Ranking (BPR+) loss.

In the graph self-supervised learning (GSL) framework, abstract nodes are introduced to reconstruct the original network, preserving existing node relationships. The Bayesian Personalized Ranking (BPR) loss evaluates user-item relationships in this restructured network based on 2-hop path statistic. Previous work has demonstrated that 2-hop paths serve as approximate representations of energy distance in the latent hyperbolic space [8]. Our research, based on the maximum entropy principle, demonstrates that weighting 2-hop paths aligns more closely with the actual energy distance, revealing BPR loss's limitations. This led to the introduction of TopoLa distance, defined as follows:

$$d_{\text{topo}}(u, i) = \frac{1}{\lambda} |2 - \text{hop}| - \frac{1}{\lambda^2} |4 - \text{hop}| + \frac{1}{\lambda^3} |6 - \text{hop}| - \dots \quad (6)$$

where  $\lambda$  is the scale factor. The BPR+ loss (Figure 3) can be represented as:

$$L_{\text{BPR+}} = - \sum_u^m \sum_{i \in \mathcal{N}_u} \sum_{j \notin \mathcal{N}_u} \ln \sigma \left( \frac{d_{\text{topo}}(u, i) - d_{\text{topo}}(u, j)}{\lambda} \right) + \tau \|\mathbf{E}^0\|^2. \quad (7)$$

This incorporation of all even-hop paths offers two advantages: (1) global information is introduced to amalgamate relational information of all users and items in the network; (2) The TopoLa distance captures the topological structure information between users and items (**Theorem 2** and **Theorem 3**). See Appendix B and C for details.

**Theorem 2.** *Given a user embedding  $y \in \mathbb{R}^{N_e}$ , the embedding matrix  $\mathbf{E} \in \mathbb{R}^{(m+n) \times N_e}$ , and a parameter  $\lambda$ , the following optimal solution to the problem is denoted by  $\mathbf{c}^*$  in vector form:*

$$\min \frac{1}{\lambda} \|\mathbf{y} - \mathbf{c}\mathbf{E}\|_F^2 + \|\mathbf{c}\|_F^2 \quad (8)$$

We have

$$\frac{\|\mathbf{c}_i^* - \mathbf{c}_j^*\|_F^2}{\|\mathbf{y}\|_F^2} \leq \frac{1}{\lambda} \|\mathbf{e}_i^\top - \mathbf{e}_j^\top\|_F^2 \quad (9)$$

**Theorem 3.** *Given nodes  $i$  and  $j$ , the topological similarity between them is directly proportional to the  $d_{\text{topo}}$  value.*

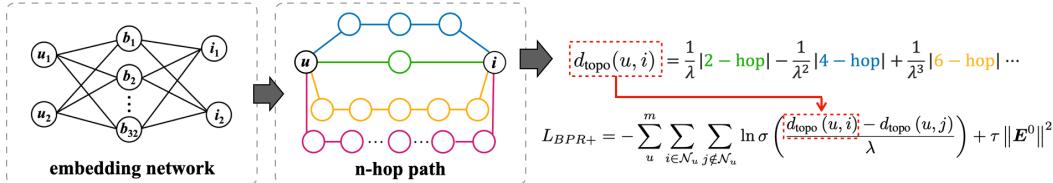


Figure 3: The illustration of BPR+ loss.

## 6 Optimizing loss computation

Direct computation of the loss using the TopoLa distance from the previous section, which involves multiple matrix multiplications, is computationally inefficient. To address this, we have optimized the computational approach to enhance efficiency. The matrix power form of the TopoLa distance can be defined as:

$$\mathbf{D}_{\text{topo}} = \frac{1}{\lambda} \mathbf{H} - \frac{1}{\lambda^2} \mathbf{H}^2 + \frac{1}{\lambda^3} \mathbf{H}^3 - \frac{1}{\lambda^4} \mathbf{H}^4 \dots \quad (10)$$

where  $\mathbf{H} = \mathbf{E}\mathbf{E}^\top$  is the 2-hop path statistic matrix of  $\mathbf{E}$ .  $d_{\text{topo}}(u, i)$  is an element of  $\mathbf{D}_{\text{topo}}$ , denoted as  $(\mathbf{D}_{\text{topo}})_{ui} = d_{\text{topo}}(u, i)$ . Due to the necessity of multiple matrix multiplications, the time complexity for calculating matrix  $\mathbf{D}_{\text{topo}}$  is  $O(N_b^3) \cdot N_b$  is the batch size. To enhance computational efficiency, we employ matrix decomposition techniques. Firstly, the embedding matrix  $\mathbf{E}$  undergoes singular value decomposition (SVD), represented as  $\mathbf{E} = \mathbf{U}\Sigma\mathbf{V}^\top$ . Consequently, the formula for  $\mathbf{D}_{\text{topo}}$  can be expressed as:

$$\mathbf{D}_{\text{topo}} = \mathbf{U} \left( \frac{1}{\lambda} \Sigma^2 - \frac{1}{\lambda^2} \Sigma^4 + \frac{1}{\lambda^3} \Sigma^6 - \frac{1}{\lambda^4} \Sigma^8 \dots \right) \mathbf{U}^\top. \quad (11)$$

As a result, the time complexity is reduced to  $\mathcal{O}(N_b N_e^2 + N_b^2 N_e + N_e^3)$ .  $N_e$  is the embedding size. Given that in GSL models,  $N_b$  significantly exceeds  $N_e$ , the proposed calculation form offers a lower time complexity.

## 7 Experiments

To confirm the superiority and efficacy of the proposed BPR+ loss, we carried out extensive experiments to explore the following research questions:

- **RQ1:** How much improvement can our proposed loss bring when applied to various state-of-the-art GSL models?
- **RQ2:** How do the physical properties of our proposed loss differ from those of Bayesian Personalized Ranking (BPR) loss?
- **RQ3:** How can numerical analysis enhance the efficiency of loss computation?
- **RQ4:** How does our proposed technique perform in practical applications?

### 7.1 Experimental Settings

**Evaluation Datasets.** In this section, we evaluate our model and several baselines on five real-world datasets: Amazon, Gowalla, Yelp, LastFM, and Beer. The Amazon dataset includes implicit feedback from users on books from the Amazon platform. The Gowalla dataset, derived from the Gowalla platform, records user check-ins at different locations, with data provided for January to June 2010. The Yelp dataset contains user ratings and interaction data for various businesses. The LastFM dataset comprises social network interactions, tags, and music artist listening information from the Last.fm online music system. The Beer dataset, sourced from the BeerAdvocate platform, includes beer reviews and is filtered to only retain users and items with at least ten interactions. Details and statistics are provided in Table 1.

Table 1: Statistics of the experimental datasets.

Dataset	User #	Item #	Interaction #	Density
Amazon	76,469	83,761	966,680	$1.5 \times 10^{-4}$
Gowalla	25,557	19,747	294,983	$5.8 \times 10^{-4}$
Yelp	42,712	26,822	182,357	$1.6 \times 10^{-4}$
LastFM	1,892	17,632	92,834	$2.8 \times 10^{-3}$
Beer	10,456	13,845	1,381,094	$9.5 \times 10^{-3}$

**Evaluation Protocols.** We downloaded the datasets from prior work, including its predefined training, validation, and test sets. The evaluation metrics used include Recall@N and Normalized Discounted Cumulative Gain (NDCG)@N, where  $N = \{10, 20\}$ , both of which are better when higher.

**Baseline Methods.** We evaluated the general applicability of the proposed technique by applying BPR+ loss to various baseline models. The details of these baselines are provided below.

- LightGCN [9]: It simplifies the traditional Graph Convolution Network (GCN) for collaborative filtering by focusing solely on neighborhood aggregation, eschewing feature transformations and nonlinear activations to enhance training efficiency and improve recommendation performance.

- SGL [10]: It improves LightGCN by using contrastive learning with data augmentation techniques like random walks and node/edge dropout, enhancing accuracy and robustness.
- NCL [11]: It is a neighborhood-enriched contrastive learning approach that boosts the effectiveness of graph collaborative filtering by incorporating structural and semantic neighbors into contrastive pairs, enhancing the model’s ability to handle data sparsity and improve recommendation accuracy.
- LightGCL [12]: It is a graph contrastive learning approach that enhances graph-based recommender systems by using singular value decomposition for data augmentation, effectively preserving the intrinsic semantic structures and offering robustness against noise perturbation and data sparsity.
- AdaGCL [5]: It is a graph contrastive learning framework that improves collaborative filtering through two trainable view generators: a graph generative model and a graph denoising model, which create adaptive contrastive views to effectively address data sparsity and noise issues.

## 7.2 Overall Performance Validation (RQ1)

The effectiveness of the proposed BPR+ loss is validated by applying it to various baseline models and conducting a comprehensive performance evaluation across five datasets. We present the experimental result in Table 2. The findings indicate a consistent pattern: algorithms incorporating BPR+ loss exhibit improved performance across all datasets and metrics. For instance, in the Amazon dataset, compared to adaGCL, adaGCL+ achieves an 11.9% improvement in the Recall@10 metric; for the NDCG@10 metric, the increase reaches 13.4%. The consistent improvement across almost all datasets and algorithms highlights the versatility and robustness of the BPR+ loss. This enhancement not only leads to higher recall rates, meaning more relevant items are recommended, but also improves the NDCG scores, indicating a better ranking quality in the recommendations provided. See Appendix D for details. In addition, see Appendix E for hyperparameter analysis.

Table 2: Performance comparison on Amazon, Gowalla, Yelp, LastFM, Beer datasets in terms of Recall and NDCG.

Dataset	Metric	adaGCL	adaGCL+	lightGCL	lightGCL+	NCL	NCL+	SGL	SGL+	lightGCN	lightGCN+
Amazon	Recall@10	0.0612	<b>0.0685</b>	0.0576	<b>0.0608</b>	0.0736	<b>0.0783</b>	0.0432	<b>0.0454</b>	0.0535	<b>0.0722</b>
	NDCG@10	0.0507	<b>0.0575</b>	0.0476	<b>0.0510</b>	0.0608	<b>0.0656</b>	0.0269	<b>0.0282</b>	0.0439	<b>0.0607</b>
	Recall@20	0.0946	<b>0.1032</b>	0.0892	<b>0.0942</b>	0.1125	<b>0.1166</b>	0.0678	<b>0.0690</b>	0.0838	<b>0.1080</b>
	NDCG@20	0.0620	<b>0.0691</b>	0.0583	<b>0.0621</b>	0.0739	<b>0.0784</b>	0.0343	<b>0.0351</b>	0.0541	<b>0.0727</b>
Gowalla	Recall@10	0.1523	<b>0.1536</b>	0.1690	<b>0.1693</b>	0.1755	<b>0.1768</b>	0.1641	<b>0.1650</b>	0.1544	<b>0.1564</b>
	NDCG@10	0.1206	<b>0.1208</b>	0.1359	<b>0.1365</b>	0.1409	<b>0.1420</b>	0.1321	<b>0.1323</b>	0.1248	<b>0.1264</b>
	Recall@20	0.2253	<b>0.2264</b>	0.2444	<b>0.2451</b>	0.2557	<b>0.2571</b>	0.2369	<b>0.2369</b>	0.2297	<b>0.2317</b>
	NDCG@20	0.1427	<b>0.1432</b>	0.1585	<b>0.1592</b>	0.1649	<b>0.1663</b>	0.1540	<b>0.1541</b>	0.1474	<b>0.1489</b>
Yelp	Recall@10	0.0517	<b>0.0529</b>	0.0456	<b>0.0520</b>	0.0503	<b>0.0510</b>	0.0654	<b>0.0739</b>	0.0505	<b>0.0525</b>
	NDCG@10	0.0320	<b>0.0334</b>	0.0284	<b>0.0330</b>	0.0313	<b>0.0318</b>	0.0554	<b>0.0742</b>	0.0314	<b>0.0325</b>
	Recall@20	0.0805	<b>0.0837</b>	0.0716	<b>0.0830</b>	<b>0.0794</b>	0.0790	0.0978	<b>0.1055</b>	<b>0.0798</b>	0.0796
	NDCG@20	0.0406	<b>0.0425</b>	0.0362	<b>0.0421</b>	0.0398	<b>0.0400</b>	0.0662	<b>0.0863</b>	0.0400	<b>0.0405</b>
LastFM	Recall@10	0.1688	<b>0.1760</b>	0.1403	<b>0.1412</b>	0.1772	<b>0.1775</b>	0.1655	<b>0.1734</b>	0.1511	<b>0.1715</b>
	NDCG@10	0.1521	<b>0.1553</b>	<b>0.1243</b>	0.1239	0.1562	<b>0.1566</b>	0.1470	<b>0.1526</b>	0.1343	<b>0.1529</b>
	Recall@20	0.2465	<b>0.2499</b>	0.1949	<b>0.1964</b>	0.2525	<b>0.2538</b>	0.2433	<b>0.2486</b>	0.2206	<b>0.2480</b>
	NDCG@20	0.1843	<b>0.1860</b>	<b>0.1475</b>	0.1472	0.1877	<b>0.1883</b>	0.1790	<b>0.1836</b>	0.1634	<b>0.1847</b>
Beer	Recall@10	0.0448	<b>0.0457</b>	0.0320	<b>0.0324</b>	0.0729	<b>0.0734</b>	0.0730	<b>0.0731</b>	0.0670	<b>0.0757</b>
	NDCG@10	0.0584	<b>0.0608</b>	0.0533	<b>0.0538</b>	0.0922	<b>0.0923</b>	0.0922	0.0922	0.0857	<b>0.0937</b>
	Recall@20	0.0731	<b>0.0757</b>	0.0489	<b>0.0491</b>	0.1171	<b>0.1175</b>	0.1165	<b>0.1167</b>	0.1084	<b>0.1227</b>
	NDCG@20	0.0647	<b>0.0675</b>	0.0542	<b>0.0547</b>	0.1013	<b>0.1014</b>	0.1009	0.1009	0.0938	<b>0.1039</b>

## 7.3 Physical Property Analysis (RQ2)

In this section, we explore the physical properties of BPR loss and BPR+ loss. To simulate ID embeddings, a  $1000 \times 64$  matrix is constructed with its elements set to 0 and 1. Our first evaluation focused on whether these loss functions could accurately reflect the topological similarity between users and items, as calculated using the Jaccard index. Observations from Figures 4A and 4B demonstrate that the relationships derived by both loss functions proportionally align with topological similarities. However, BPR exhibits a coarse granularity, where different topological similarities may correspond to identical scores.

Further analysis involved examining the relationship between the number of neighbors in a user-item pair and the relationship scores. The number of neighbors indicates the norm of the user/item embeddings. The union of neighbors represents the number of non-zero elements in the sum of two

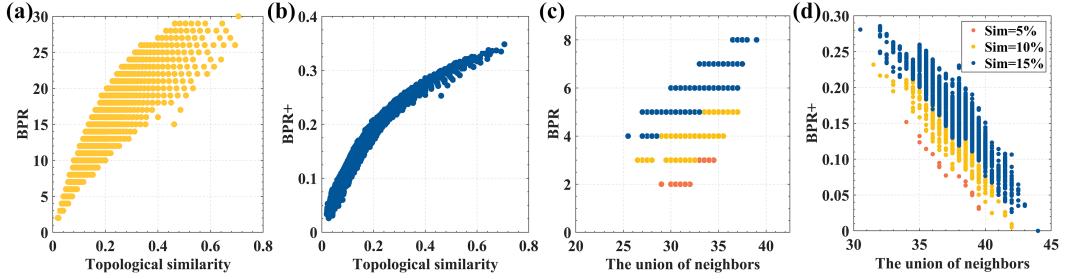


Figure 4: Comparison of the physical properties of BPR loss and BPR+ loss.

embedding vectors. A user connected to numerous abstract nodes will have a larger norm, leading to higher dot products with all item embeddings. Our experimental validation supports this analysis: The BPR loss scores relationships, increasing with the expanding union of neighbors. Additionally, when classifying these scores by varying degrees of topological similarity (within a range of  $\pm 2\%$ ), we observe that BPR delineates less distinct boundaries between different similarity levels with a narrower range of scores. In contrast, BPR+ offers a more nuanced evaluation, assigning distinct scores to different degrees of similarity, thus enhancing the fidelity of relationship assessments.

#### 7.4 Efficiency Study (RQ3)

Due to the inefficiency of calculating loss directly via matrix exponentiation, we incorporate matrix factorization techniques to accelerate the loss computation process. To analyze computational efficiency in practical scenarios, we select the NCL as an example and compared the computation times of three loss computation methods: BPR, BPR+ (MM), and BPR+ (MF), across five datasets, as illustrated in Figure 5A. We record the time required to complete one epoch for each loss function. BPR+ (MM) denotes BPR+ calculated using matrix multiplication, whereas BPR+ (MF) refers to BPR+ calculated using matrix factorization, both considering 40-hop paths. The results indicate that using the optimized loss computation methods significantly reduces computation time, comparable to that of BPR. For instance, on the Amazon dataset, the computation time for BPR is 30.5s, for BPR+(MF) it is 36.5s, and for BPR+(MM), it significantly increases to 227.7s. Moreover, the increase in data scale further enhances the improvement in computational efficiency. Similarly, the results for LightGCL on the same datasets are consistent (Figure 5B).

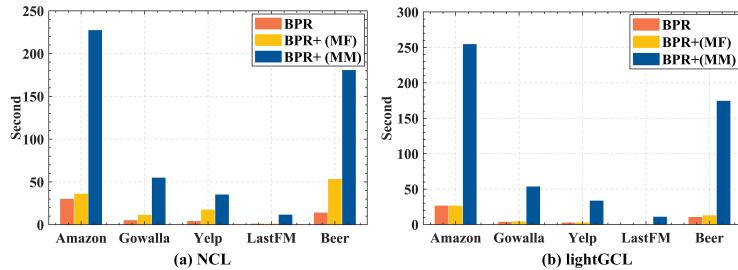


Figure 5: Comparison of computational efficiency between BPR, BPR+(MM), and BPR+(MF).

#### 7.5 Case Study (RQ4)

In this section, we apply our proposed technique specifically to the domain of drug repositioning to validate its performance in practical scenarios. We develop a new framework to aid pharmaceutical research and development (TopoDR), integrating multimodal drug and disease information encompassing chemical structures, side effects, drug-drug interactions, drug target profiles, disease phenotypes, and disease ontologies.

To verify the validity of the framework, we compare TopoDR with four leading drug repositioning methods (deepDR [13], DRHGNC [14], DDAGDL [15], and AdaDR [16]) using their default settings

from public codes, validated on gold standard datasets (Fdataset and Cdataset). Performance was assessed via 10-fold cross-validation, with results detailed in Figure 6. We use six evaluation metrics to compare the performance of TopoDR with other methods: the area under the receiver operating characteristic (ROC) curve (AUC), the area under the precision-recall curve (AUPR), accuracy (Acc.), precision (Pre.), recall (Rec.), and Matthew’s correlation coefficient (MCC). In these comparisons, particularly on the Fdataset, TopoDR consistently outperformed the other models in metrics assessing overall model performance. Specifically, in the Fdataset, TopoDR’s AUC, AUPR, and MCC are 0.9594, 0.9630, and 0.7440, respectively, surpassing AdaDR, the next best method, by 2.89%, 1.86%, and 7.34%. Similarly, on the Cdataset, TopoDR exceeds AdaDR by 1.82%, 1.18%, and 2.64%. These results confirm TopoDR’s superior ability to accurately predict potential indications for drugs. Furthermore, we select ten potential therapeutic candidates for four prevalent cancers (colorectal, breast, stomach cancer, and leukemia). Table 6 displays the top ten candidate drugs for these cancers, with confirmed therapeutic drugs from the Comparative Toxicogenomics Database (CTD) emphasized in bold. See Appendix F for details.

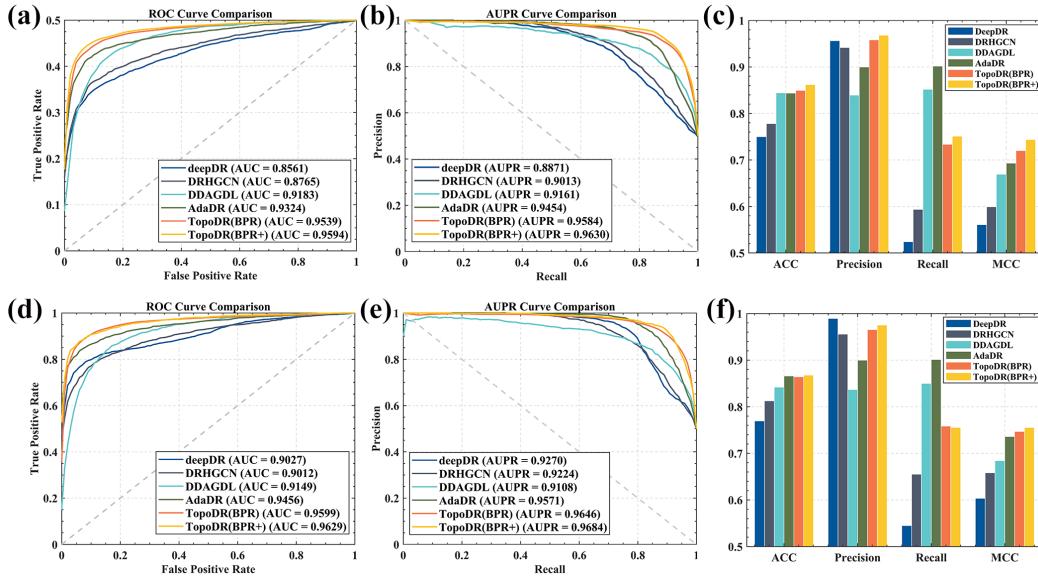


Figure 6: The performance of drug repositioning methods on Fdataset and Cdataset. Fdataset: (A) The ROC curve. (B) The precision-recall curve. (C) The accuracy, precision, recall, and Matthew’s correlation coefficient of all methods. Cdataset: (D) The ROC curve. (E) The precision-recall curve. (F) The accuracy, precision, recall, and Matthew’s correlation coefficient of all methods.

## 8 Conclusion

Although Bayesian Personalized Ranking (BPR) loss has been widely applied in graph self-supervised learning with considerable success, it currently lacks robust theoretical support and analysis for node relations evaluation. In this paper, we explain from a graph-theoretical perspective that BPR essentially counts the 2-hop paths from users to items within the embedding network. Analysis based on the principle of maximum entropy reveals that while BPR loss can represent the energy distance between users and items, it is not sufficiently precise. Based on these limitations, we propose a new loss function, BPR+, as an alternative. Our work not only elucidates the success of previous graph self-supervised learning efforts but also provides a fresh perspective for theoretical analysis in this field.

This work holds broad prospects. Foremost and most challenging is the inspiration derived from our theoretical analysis to design superior graph neural network modules. Additionally, our work represents a step towards a deeper understanding of the relationships between nodes, which is central to the utility of graph neural networks. However, computational time remains an unresolved issue. Therefore, accelerating computation will be a crucial direction for our future work.

## 9 Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (62350004, 62332020, U22A2041, 62502540, 6250072467), Shandong Provincial Natural Science Foundation (ZR2025QC1546, ZR2025QC1540), Fundamental Research Funds for the Central Universities (25CX06033A), the Project of Xiangjiang Laboratory (No. 23XJ01011), the Science and Technology Major Project of Changsha (No.kh2402004), Hunan Provincial Postgraduate Scientific Research Innovation Project (CX20230234), the start-up funds from USTC and IAI.

## References

- [1] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.
- [2] Yixin Liu, Ming Jin, Shirui Pan, Chuan Zhou, Yu Zheng, Feng Xia, and S Yu Philip. Graph self-supervised learning: A survey. *IEEE transactions on knowledge and data engineering*, 35(6):5879–5900, 2022.
- [3] Xiangnan He, Zhankui He, Xiaoyu Du, and Tat-Seng Chua. Adversarial personalized ranking for recommendation. In *The 41st International ACM SIGIR conference on research & development in information retrieval*, pages 355–364, 2018.
- [4] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618*, 2012.
- [5] Yili Wang, Kaixiong Zhou, Rui Miao, Ninghao Liu, and Xin Wang. Adagcl: Adaptive subgraph contrastive learning to generalize large-scale graph training. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 2046–2055, 2022.
- [6] Marian Boguna, Ivan Bonamassa, Manlio De Domenico, Shlomo Havlin, Dmitri Krioukov, and M Ángeles Serrano. Network geometry. *Nature Reviews Physics*, 3(2):114–135, 2021.
- [7] Dmitri Krioukov. Clustering implies geometry in networks. *Physical review letters*, 116(20):208302, 2016.
- [8] Kai Zheng, Qilong Feng, Yaohang Li, Qichang Zhao, Jinhui Xu, and Jianxin Wang. Topola: a novel embedding framework for understanding complex networks. *arXiv preprint arXiv:2405.16928*, 2024.
- [9] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. Lightgen: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 639–648, 2020.
- [10] Jiancan Wu, Xiang Wang, Fuli Feng, Xiangnan He, Liang Chen, Jianxun Lian, and Xing Xie. Self-supervised graph learning for recommendation. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, pages 726–735, 2021.
- [11] Zihan Lin, Changxin Tian, Yupeng Hou, and Wayne Xin Zhao. Improving graph collaborative filtering with neighborhood-enriched contrastive learning. In *Proceedings of the ACM web conference 2022*, pages 2320–2329, 2022.
- [12] Xuheng Cai, Chao Huang, Lianghao Xia, and Xubin Ren. Lightgcl: Simple yet effective graph contrastive learning for recommendation. *arXiv preprint arXiv:2302.08191*, 2023.
- [13] Xiangxiang Zeng, Siyi Zhu, Xiangrong Liu, Yadi Zhou, Ruth Nussinov, and Feixiong Cheng. deepdr: a network-based deep learning approach to in silico drug repositioning. *Bioinformatics*, 35(24):5191–5198, 2019.

[14] Lijun Cai, Changcheng Lu, Junlin Xu, Yajie Meng, Peng Wang, Xiangzheng Fu, Xiangxiang Zeng, and Yansen Su. Drug repositioning based on the heterogeneous information fusion graph convolutional network. *Briefings in bioinformatics*, 22(6):bbab319, 2021.

[15] Bo-Wei Zhao, Xiao-Rui Su, Peng-Wei Hu, Yu-Peng Ma, Xi Zhou, and Lun Hu. A geometric deep learning framework for drug repositioning over heterogeneous information networks. *Briefings in Bioinformatics*, 23(6):bbac384, 2022.

[16] Xinliang Sun, Xiao Jia, Zhangli Lu, Jing Tang, and Min Li. Drug repositioning with adaptive graph convolutional networks. *Bioinformatics*, 40(1):btad748, 2024.

[17] Can-Yi Lu, Hai Min, Zhong-Qiu Zhao, Lin Zhu, De-Shuang Huang, and Shuicheng Yan. Robust and efficient subspace segmentation via least squares regression. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7–13, 2012, Proceedings, Part VII 12*, pages 347–360. Springer, 2012.

[18] Xubin Ren, Lianghao Xia, Yuhao Yang, Wei Wei, Tianle Wang, Xuheng Cai, and Chao Huang. Ssrec: A self-supervised learning framework for recommendation. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, pages 567–575, 2024.

[19] Ted T Ashburn and Karl B Thor. Drug repositioning: identifying and developing new uses for existing drugs. *Nature reviews Drug discovery*, 3(8):673–683, 2004.

[20] Natalia Novac. Challenges and opportunities of drug repositioning. *Trends in pharmacological sciences*, 34(5):267–272, 2013.

[21] Vijay P Kale, Hasan Habib, Robert Chitren, Milan Patel, Kartick C Pramanik, Subash C Jonnalagadda, Kishore Challagundla, and Manoj K Pandey. Old drugs, new uses: Drug repurposing in hematological malignancies. In *Seminars in cancer biology*, volume 68, pages 242–248. Elsevier, 2021.

[22] Vijay Pralhad Kale, Shantu G Amin, and Manoj K Pandey. Targeting ion channels for cancer therapy by repurposing the approved drugs. *Biochimica et Biophysica Acta (BBA)-Biomembranes*, 1848(10):2747–2755, 2015.

[23] Joel T Dudley, Tarangini Deshpande, and Atul J Butte. Exploiting drug–disease relationships for computational drug repositioning. *Briefings in bioinformatics*, 12(4):303–311, 2011.

[24] Hanqing Xue, Jie Li, Haozhe Xie, and Yadong Wang. Review of drug repositioning approaches and resources. *International journal of biological sciences*, 14(10):1232, 2018.

[25] Nansu Zong, Andrew Wen, Sungrim Moon, Sunyang Fu, Liwei Wang, Yiqing Zhao, Yue Yu, Ming Huang, Yanshan Wang, Gang Zheng, et al. Computational drug repurposing based on electronic health records: a scoping review. *NPJ digital medicine*, 5(1):77, 2022.

[26] Xiao Liu, Fanjin Zhang, Zhenyu Hou, Li Mian, Zhaoyu Wang, Jing Zhang, and Jie Tang. Self-supervised learning: Generative or contrastive. *IEEE transactions on knowledge and data engineering*, 35(1):857–876, 2021.

[27] Ming Gao, Leihui Chen, Xiangnan He, and Aoying Zhou. Bine: Bipartite network embedding. In *The 41st international ACM SIGIR conference on research & development in information retrieval*, pages 715–724, 2018.

[28] Assaf Gottlieb, Gideon Y Stein, Eytan Ruppin, and Roded Sharan. Predict: a method for inferring novel drug indications with application to personalized medicine. *Molecular systems biology*, 7(1):496, 2011.

[29] David S Wishart, Yannick D Feunang, An C Guo, Elvis J Lo, Ana Marcu, Jason R Grant, Tanvir Sajed, Daniel Johnson, Carin Li, Zinat Sayeeda, et al. Drugbank 5.0: a major update to the drugbank database for 2018. *Nucleic acids research*, 46(D1):D1074–D1082, 2018.

[30] Ada Hamosh, Alan F Scott, Joanna S Amberger, Carol A Bocchini, and Victor A McKusick. Online mendelian inheritance in man (omim), a knowledgebase of human genes and genetic disorders. *Nucleic acids research*, 33(suppl\_1):D514–D517, 2005.

- [31] Cheng Yan and Guihua Duan. Pmdags: predicting mirna-disease associations with graph nonlinear diffusion convolution network and similarities. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2024.
- [32] Huimin Luo, Jianxin Wang, Min Li, Junwei Luo, Xiaoqing Peng, Fang-Xiang Wu, and Yi Pan. Drug repositioning based on comprehensive similarity measures and bi-random walk algorithm. *Bioinformatics*, 32(17):2664–2671, 2016.
- [33] Mengyun Yang, Lan Huang, Yunpei Xu, Chengqian Lu, and Jianxin Wang. Heterogeneous graph inference with matrix completion for computational drug repositioning. *Bioinformatics*, 36(22-23):5456–5464, 2020.
- [34] David Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences*, 28(1):31–36, 1988.
- [35] Christoph Steinbeck, Yongquan Han, Stefan Kuhn, Oliver Horlacher, Edgar Luttmann, and Egon Willighagen. The chemistry development kit (cdk): An open-source java library for chemo-and bioinformatics. *Journal of chemical information and computer sciences*, 43(2):493–500, 2003.
- [36] Philip Resnik. Using information content to evaluate semantic similarity in a taxonomy. *arXiv preprint cmp-lg/9511007*, 1995.
- [37] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864, 2016.
- [38] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. Line: Large-scale information network embedding. In *Proceedings of the 24th international conference on world wide web*, pages 1067–1077, 2015.
- [39] Daixin Wang, Peng Cui, and Wenwu Zhu. Structural deep network embedding. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1225–1234, 2016.
- [40] Chunyu Zhang, Chao Xu, Xueyun Gao, and Qingqiang Yao. Platinum-based drugs for cancer therapy and anti-tumor strategies. *Theranostics*, 12(5):2115, 2022.
- [41] Allan Peter Davis, Cynthia J Grondin, Robin J Johnson, Daniela Sciaky, Jolene Wiegers, Thomas C Wiegers, and Carolyn J Mattingly. Comparative toxicogenomics database (ctd): update 2021. *Nucleic acids research*, 49(D1):D1138–D1143, 2021.
- [42] M Riis. Modern surgical treatment of breast cancer. *Annals of medicine and surgery*, 56:95–107, 2020.
- [43] Frederic Buxant, Nadège Kindt, Guy Laurent, Jean-Christophe Noël, and Sven Saussez. Antiproliferative effect of dexamethasone in the mcf-7 breast cancer cell line. *Molecular Medicine Reports*, 12(3):4051–4054, 2015.
- [44] Jian Zhang, Zhonghua Wang, Xichun Hu, Biyun Wang, Leiping Wang, Wentao Yang, Yang Liu, Guangyu Liu, Genhong Di, Zhen Hu, et al. Cisplatin and gemcitabine as the first line therapy in metastatic triple negative breast cancer. *International journal of cancer*, 136(1):204–211, 2015.

## A Proof of Theorem 1

We restate Theorem 1: *Given a network  $X \in \mathbb{R}^{(m+n) \times N_e}$ , let  $\langle t(x) \rangle$  denote the expected number of triangles, and  $\langle t'(x) \rangle$  represent the expected number of weighted triangles. Consequently,  $\alpha' > \alpha$ , where  $\alpha'$  is the logarithm of the thermodynamic activity corresponding to  $\langle t'(x) \rangle$ , and  $\alpha$  is associated with  $\langle t(x) \rangle$ .*

**Proof.** In network geometry, the expected numbers of triangles can be obtained through the integration of the graphon [7] :

$$\langle t(x) \rangle = \frac{1}{2} \iint_{\mathbb{R}^2} p(x, y)p(y, z)p(z, x)dydz, \quad (12)$$

Under the assumption of redundancy in the current statistics, the non-redundant expectation  $\langle t'(x) \rangle$ , is anticipated to be less than its redundant counterpart,  $\langle t(x) \rangle$ . In this context,  $t(x)$  denotes the quantity of triangles associated with node  $x$ . Utilizing the maximum-entropy principle (21), we derive the following formula:

$$\langle t'(x) \rangle = \frac{1}{2} \iint_{\mathbb{R}^2} p'(x, y)p'(y, z)p'(z, x)dydz = \bar{t}' < \bar{t}. \quad (13)$$

Therefore,

$$\frac{1}{2} \iint_{\mathbb{R}^2} p'(x, y)p'(y, z)p'(z, x)dydz < \frac{1}{2} \iint_{\mathbb{R}^2} p(x, y)p(y, z)p(z, x)dydz, \quad (14)$$

When the network size is sufficiently large, the approximate solution for the graphon that maximizes entropy is the Fermi-Dirac graphon:

$$p^*(x, y) = \begin{cases} \frac{1}{1+e^{2\alpha(r-\frac{1}{2})}} & \text{if } 0 \leq r \leq 1, \\ \frac{1}{1+e^\alpha} & \text{if } r > 1, \end{cases} \quad (15)$$

where  $\alpha$  and  $r$  are the rescaled inverse temperature and energy distance, respectively. Inserting the terms from Formula [17] into Formula [16], we obtain:

$$\iint_{\mathbb{R}^2} \left[ \frac{1}{1+e^{2\alpha(r-\frac{1}{2})}} \right]^3 dydz < \iint_{\mathbb{R}^2} \left[ \frac{1}{1+e^{2\alpha(r-\frac{1}{2})}} \right]^3 dydz. \quad (16)$$

Consequently, we deduce that  $\alpha' > \alpha$ . Based on prior research, the common neighbor integral corresponding to  $\alpha'$  provides a more accurate representation of the latent space distance than that associated with  $\alpha$  [7]. From this, we demonstrate that the current triangle statistics exhibit redundancy. Given that the common-neighbor integral is defined as  $\int_{\mathbb{R}} \left[ \frac{1}{1+e^{2\alpha(r-\frac{1}{2})}} \right]^2 dz$ , the 2-hop paths (common neighbors) are also found to be imprecise when representing distances within the latent space and **Theorem 1** holds.

## B Proof of Theorem 2

We restate Theorem 2: *Given a user embedding  $y \in \mathbb{R}^{N_e}$ , the embedding matrix  $\mathbf{E} \in \mathbb{R}^{(m+n) \times N_e}$ , and a parameter  $\lambda$ , the following optimal solution to the problem is denoted by  $\mathbf{c}^*$  in vector form:*

$$\min \frac{1}{\lambda} \|\mathbf{y} - \mathbf{c}\mathbf{E}\|_F^2 + \|\mathbf{c}\|_F^2 \quad (17)$$

We have

$$\frac{\|\mathbf{c}_i^* - \mathbf{c}_j^*\|_F^2}{\|\mathbf{y}\|_F^2} \leq \frac{1}{\lambda} \|\mathbf{e}_i^\top - \mathbf{e}_j^\top\|_F^2 \quad (18)$$

**Proof.** Inspired by prior work, we conducted the following inference [17]. Let  $L(c) = \frac{1}{\lambda} \|\mathbf{y} - \mathbf{c}\mathbf{E}\|_F^2 + \|\mathbf{c}\|_F^2$ . Since  $\mathbf{c}^*$  is the optimal solution to Equation (17), it satisfies

$$\frac{\partial L(c)}{\partial c_k} \Big|_{c=c^*} = 0 \quad (19)$$

Thus, we have

$$-\frac{2}{\lambda} \mathbf{e}_i^\top (\mathbf{y} - \mathbf{c}^* \mathbf{E}) + 2\mathbf{c}_i^* = 0, \quad (20)$$

$$-\frac{2}{\lambda} \mathbf{e}_j^\top (\mathbf{y} - \mathbf{c}^* \mathbf{E}) + 2\mathbf{c}_j^* = 0. \quad (21)$$

Equation (20) and (21) give us

$$\mathbf{c}_i^* - \mathbf{c}_j^* = \frac{1}{\lambda} (\mathbf{e}_i^\top - \mathbf{e}_j^\top) (\mathbf{y} - \mathbf{c}^* \mathbf{E}). \quad (22)$$

Since  $\mathbf{c}^*$  is optimal to equation (17), we get

$$\frac{1}{\lambda} \|\mathbf{y} - \mathbf{c}^* \mathbf{E}\|_F^2 + \|\mathbf{c}^*\|_F^2 = L(\mathbf{c}^*) \leq L(0) = \|\mathbf{y}\|_F^2 \quad (23)$$

Thus, we have  $\frac{1}{\lambda} \|\mathbf{y} - \mathbf{c}^* \mathbf{E}\|_F^2 < \|\mathbf{y}\|_F^2$ . Then equation (23) implies

$$\frac{\|\mathbf{c}_i^* - \mathbf{c}_j^*\|_F^2}{\|\mathbf{y}\|_F^2} \leq \frac{1}{\lambda} \|\mathbf{e}_i^\top - \mathbf{e}_j^\top\|_F^2 \quad (24)$$

The optimal solution  $\mathbf{c}_i$  of optimization problem (17) corresponds to the  $i$ -th row of matrix  $\mathbf{D}_{\text{topo}}$ . From the above equation, we discern that the difference in distances between two nodes in the latent space relative to other nodes correlates with the divergence in their topological structures. That is, nodes with highly similar topological structures occupy proximate positions within the latent space and **Theorem 2** holds.

## C Proof of Theorem 3

We restate Theorem 3: *Given nodes  $i$  and  $j$ , the topological similarity between them is directly proportional to the  $d_{\text{topo}}$  value.*

**Proof.** Our way to exploring the relationship between Topola distance and the topological similarity is to differentiate various types of paths connecting nodes  $i$  and  $j$ . By counting the number of each type of paths, it allows us to take into consideration not only local and global connectivity, but also node degrees in an implicit manner. Particularly, we focus on the classification of  $n$ -hop paths connecting nodes  $i$  and  $j$ . Due to the possible existence of loop or loops on these paths, we establish a classification system for all such  $n$ -hop paths, based on the length of a path actually traversed (i.e., the number of hops after removing all the loops on the path). Clearly, there are a total of  $(n-1)$  types. To facilitate a clear differentiation, we represent these path types with polygons. For example, a path that actually traverses 2-hop is defined as a  $P_2$  path, which can be viewed as a triangle if adding a direct edge back from node  $j$  to node  $i$ . Similarly, a path that actually traverses 3 hops can be defined as  $P_3$  (represented as a quadrangle), and a path that actually traverses 4 hops can be defined as  $P_4$  (represented as a pentagon). The quantity of these types can be expressed as  $|P_l| = \text{loop}(a_l)$ , where  $a_l$  is the set of loop-free  $l$ -hop paths connecting nodes  $i$  and  $j$  and  $\text{loop}(\cdot)$  represents the number of different ways of adding loops to paths in  $a_l$  to form  $n$ -hop paths between nodes  $i$  and  $j$ . While there is a correlation between  $|P_l|$  and  $|a_l|$ , accurately measuring this relationship remains challenging. Nevertheless, our observations suggest that a proportional relationship exists between the quantity of  $P_l$ , characterized by solely loops between node  $i$  and its neighbors, or between node  $j$  and its neighbors, and  $|a_l|$ . Paths exhibiting such characteristics are defined as  $b_n(l)$ . The interplay between  $|a_l|$  and  $|b_n(l)|$  can be calculated as follows:

$$|b_n(l)| = \sum_{h=0}^{\frac{n-l}{2}} \kappa_i^h \kappa_j^{\frac{n-l}{2}-h} |a_l| \quad (25)$$

where  $|b_n(l)|$  is the quantity of  $b_n(l)$  in  $n$ -hop  $P_l$  paths.  $\kappa_i$  and  $\kappa_j$  represent the degrees of node  $i$  and node  $j$ . Thus,  $n$ -hop paths between node  $i$  and node  $j$  in the network are represented as:

$$\begin{aligned} |n - \text{hop}| &= |a_n| + |b_n| + |c_n| \\ &= |a_n| + \sum_{P_l \in n - \text{hop}} |b_n(P_l)| + |c_n| \\ &= \begin{cases} \sum_{t=1}^{\frac{n}{2}} \sum_{h=0}^{\frac{n}{2}-t} \kappa_i^h \kappa_j^{\frac{n}{2}-t-h} |a_{2t}| + |c_n| & \text{if } n \text{ is even} \\ \sum_{t=1}^{\frac{n-1}{2}} \sum_{h=0}^{\frac{n-1}{2}-t} \kappa_i^h \kappa_j^{\frac{n-1}{2}-t-h} |a_{2t+1}| + |c_n| & \text{if } n \text{ is odd} \end{cases} \end{aligned} \quad (26)$$

where  $c_n$  denotes the paths distinct from  $a_n$  and  $b_n$ .  $|n - \text{hop}|$  refers to the counts of  $n$ -hop paths. Our analysis for 2-hop, 4-hop, and 6-hop paths reveals that  $|c_n|$  does not exhibit a direct correlation with  $\kappa_i$  and  $\kappa_j$ . Equation (26) reveals that  $a_n$  and  $b_n$  for even-hop paths consist of polygons with an odd number of sides, while for odd-hop paths, they comprise polygons with an even number of sides. Subsequently, we delve further into the physical significance of  $D_{\text{topo}}$ , which can also be expressed in the form of a set of paths (note that the following formulas reveal the physical meanings of  $D_{\text{topo}}$  and  $d_{\text{topo}}(i, j)$ , but will not be used for calculation):

$$D_{\text{topo}} = \frac{1}{\lambda} \mathbf{A} \mathbf{A}^\top - \frac{1}{\lambda^2} \mathbf{A} \mathbf{A}^\top \mathbf{A} \mathbf{A}^\top + \frac{1}{\lambda^3} \mathbf{A} \mathbf{A}^\top \mathbf{A} \mathbf{A}^\top \mathbf{A} \mathbf{A}^\top - \dots \quad (27)$$

The emphasis on even-hop paths arises from their ability to capture the degree information and the global connectivity information of nodes  $i$  and  $j$ . Furthermore, the TopoLa distance  $d_{\text{topo}}$  is demonstrated to correlate with the topological structure of nodes. Specifically,  $a_l$  can be classified into two types: those overlapping with 2-hop paths ( $g_l$ ) and the remaining ones ( $s_l$ ), with  $|a_l| = |g_l| + |s_l|$ . The  $|s_l|$  relates to the topological structure of nodes  $i$  and  $j$ . For instance, identical topological structure between  $i$  and  $j$  renders  $|s_l|$  to zero (which indicates that it measures the topological similarity of nodes  $i$  and  $j$ ). Thus, we can express  $d_{\text{topo}}$  as:

$$\begin{aligned} d_{\text{topo}}(i, j) &= \lim_{n \rightarrow \infty} \left[ \left( \sum_{t=1}^{\frac{n}{2}} \sum_{h=0}^{\frac{n}{2}-t} \frac{\kappa_i^h \kappa_j^{\frac{n}{2}-t-h}}{-(-\lambda)^{\frac{n}{2}-t+1}} \right) \left( \sum_{t=1}^{\frac{n}{2}} \frac{|a_{2t}|}{(-\lambda)^{t-1}} \right) + \sum_{t=1}^{\frac{n}{2}} \frac{c_{(2t)}}{\lambda^t} \right] \\ &= \lim_{n \rightarrow \infty} \left[ \left( \sum_{t=1}^{\frac{n}{2}} \sum_{h=0}^{\frac{n}{2}-t} \frac{\kappa_i^h \kappa_j^{\frac{n}{2}-t-h}}{-(-\lambda)^{\frac{n}{2}-t+1}} \right) \left[ \sum_{t=1}^{\frac{n}{2}} \frac{|g_{2t}|}{(-\lambda)^{t-1}} - \sum_{t=2}^{\frac{n}{2}} \frac{|s_{2t}|}{(-\lambda)^{t-1}} \right] + \sum_{t=1}^{\frac{n}{2}} \frac{c_{(2t)}}{\lambda^t} \right] \end{aligned} \quad (28)$$

where  $\sum_{t=1}^{\frac{n}{2}} \frac{|g_{2t}|}{(-\lambda)^{t-1}}$  and  $\sum_{t=1}^{\frac{n}{2}} \frac{c_{(2t)}}{\lambda^t}$  are unaffected by node's topological structure, while  $\sum_{t=2}^{\frac{n}{2}} \frac{|s_{2t}|}{(-\lambda)^{t-1}}$  acts as a topology-dependent penalty term. We find that the topological similarity between  $i$  and  $j$  is directly proportional to the  $d_{\text{topo}}$  value and **Theorem 3** holds.

## D More details about overall performance validation

The code for all models originates from SSLRec, with each model's embedding size fixed at 32 and the batch size set to 4096. To ensure a fair comparison, we employed a grid search method to determine the optimal parameter combination for each model, with  $\lambda$  search range set to {1e-3, 1e-4, 1e-5, 1e-6, 1e-7}.

The Amazon, Gowalla, and Yelp datasets were downloaded from SSLRec [18] and have been pre-divided into training, validation, and test sets. The LastFM and Beer datasets were obtained from the corresponding code and data provided by adaGCL [5], and have also been pre-partitioned.

Experiments were conducted on a high-performance hardware platform comprising an Intel Xeon Platinum 8352V processor, an NVIDIA RTX 4090 with 24 GB of memory, and a system running

Ubuntu 20.04. The software environment included PyTorch version 1.11.0, Python 3.8, and CUDA 11.3.

To ensure statistical significance, we trained models using BPR loss and BPR+ loss, calculating the p-values. The results are presented in Table 3.

Table 3: Statistical significance of overall performance comparison.

Dataset	adaGCL	lightGCL	NCL	SGL	lightGCN
Amazon	1.97E-12	1.20E-18	1.34E-08	1.31E-13	7.37E-12
Gowalla	6.13E-11	3.29E-16	1.76E-13	3.72E-19	3.80E-14
Yelp	1.99E-10	5.31E-13	4.90E-11	2.78E-10	8.82E-08
LastFM	4.24E-12	2.38E-30	9.18E-10	1.65E-14	1.74E-11
Beer	7.56E-11	3.91E-10	8.44E-14	2.19E-15	3.17E-12

## E Hyperparameter analysis

In this section, we explore the sensitivity of the adaGCL model to the critical hyperparameter  $\lambda$  in the BPR+ loss formulation, which governs the influence of n-hop paths on relationship assessments. To ascertain the optimal  $\lambda$ , we systematically search across a range  $\{1e-3, 1e-4, 1e-5, 1e-6, 1e-7\}$ , aiming to determine how it impacts adaGCL’s performance. As illustrated in Figure 7, the model’s performance across five datasets with varying  $\lambda$  values is analyzed. The data suggests that the optimal performance can occur at any point within the tested  $\lambda$  range, highlighting the varying influence of n-hop paths on different datasets. This variability underscores the need for dataset-specific tuning of  $\lambda$  to achieve the best results in adaGCL.

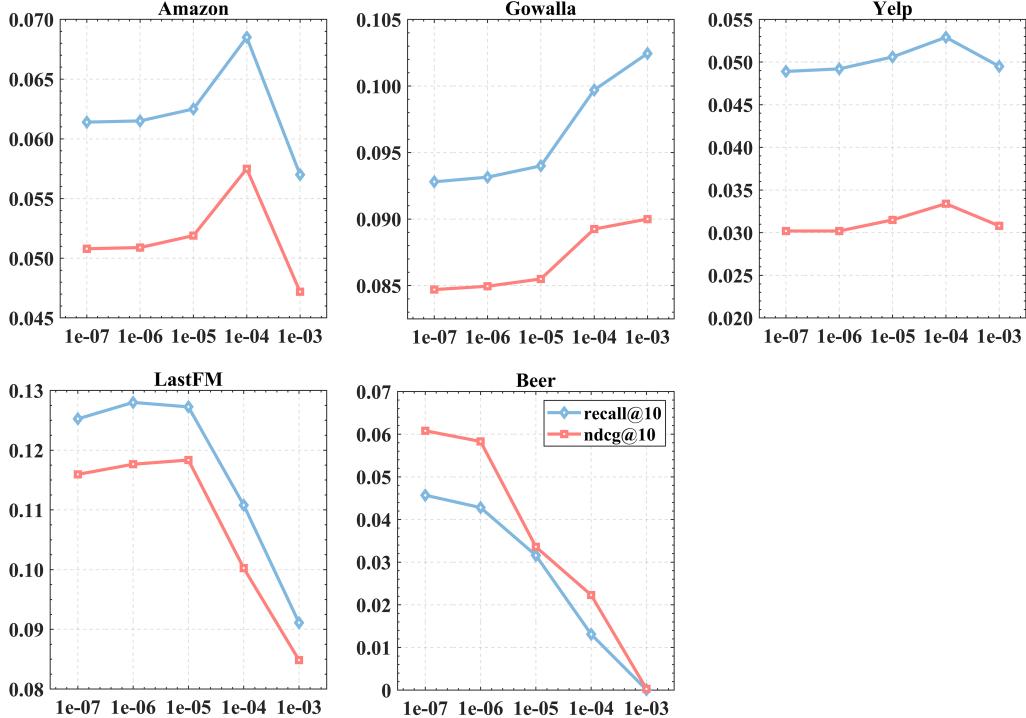


Figure 7: The impact of  $\lambda$

## F More details about case study

Drug repositioning, encompassing strategies like repositioning and therapeutic switching, is an evolving paradigm in pharmaceutical research, recognized for its time and cost efficiency compared

to traditional drug development [19, 20]. It leverages existing drugs, bypassing initial development phases, and accelerating market entry. Furthermore, drug repositioning emerges as a pragmatic way for addressing complex diseases like cancer [21, 22]. Computational drug repositioning utilizes molecular, clinical, and biophysical data to ascertain the potential of approved drugs for new clinical applications [23]. Such a strategy not only expedites the repositioning process but also minimizes associated costs [24]. Researchers have actively applied this strategy in identifying and developing therapeutic agents against COVID-19 [25].

With the advancement of artificial intelligence technology and the accumulation of large-scale biomedical data, drug repositioning based on deep learning (DL) has shown significant advantages over traditional computational approaches. For instance, DeepDR employed a collective variational autoencoder to integrate diverse network data, enhancing drug repositioning by effectively capturing non-linear structures and mitigating data sparsity [13]. Another innovative technique, DRHGCN, integrated data from various networks to refine drug repositioning [14]. It extracted both intra-domain and inter-domain features, improving drug and disease representations through sophisticated layer attention mechanisms, thereby enhancing repositioning accuracy. Additionally, DDAGDL, a framework that incorporated diverse drug-related networks, employed non-Euclidean domain knowledge and an attention mechanism to strengthen feature representation [15]. This framework further used XGBoost for robust prediction tasks. Lastly, AdaDR employed graph neural networks and attention mechanisms to integrate node features with topological structures [16]. Self-Supervised Learning (SSL) represents a potent paradigm that diminishes the dependency on manually annotated labels, facilitating training on extensive unlabeled datasets via self-supervised tasks [26]. It has achieved notable success across multiple domains, encompassing visual representation learning, pre-training of language models, and beyond [26]. Within the sphere of recommender systems, graph self-supervised learning (GSL) has gained prominence [2]. This paradigm has evolved from its initial unsupervised methods to sophisticated models that utilize network embeddings and random-walk proximity for self-supervision [27]. LightGCN, recognized for its simplicity and efficacy, has emerged as a favored graph self-supervised learning method, and forms the foundation of numerous graph contrastive learning (GCL) methods [9, 12]. Bayesian Personalized Ranking (BPR) loss, employed in LightGCN, is also a prevalent loss function in latest GCL methods [4]. However, within the context of complex networks, BPR loss relies on 2-hop path statistics for assessing sample relationships. Previous research has proved that relying solely on this local connectivity information may not suffice for accurately measuring the energy distance between nodes in the latent hyperbolic space. This limitation might neglect critical, broader network patterns, potentially limiting its effectiveness in biological analysis.

In this section, we develop a graph self-supervised learning framework using Topology-encoded Latent Hyperbolic Geometry for drug repositioning. By assigning ID embeddings to drugs and diseases and refining them across the drug-disease network, we aggregate these embeddings at various propagation layers to form the final network embedding. Multisource information about drugs and diseases is then integrated for drug repositioning tasks.

To summarize, this framework makes the following main contributions:

- We build TopoDR, a new computational drug repositioning framework based on lightGCN+, which integrates multimodal information.
- Empirical analyses on benchmark datasets reveal that our approach surpasses current drug repositioning algorithms in predictive accuracy.

## F.1 Materials and methods

In this section, we first delineate the benchmark datasets for drug repositioning, along with the multimodal representations related to drugs and diseases. Subsequently, we present the framework of the drug repositioning model, TopoDR, which primarily comprises two modules: the network embedding module and the prediction module. The primary contribution of this work lies in the network embedding module, developed through the graph self-supervised learning method, lightGCN+. This method is grounded in Topology-encoded Latent Hyperbolic Geometry. The prediction module integrates multimodal information to forecast alternative indications of drugs.

### F.1.1 Materials

The Fdataset and Cdataset are benchmark datasets in drug repositioning research. The Fdataset, recognized as the gold standard, comprises 1933 verified drug-disease associations, integrating 593 drugs from DrugBank and 313 diseases from the OMIM database [28, 29, 30, 31]. The Cdataset, in turn, widens the data collection spectrum, encompassing 663 drugs, 409 diseases, and 2352 drug-disease associations [32]. Together, these datasets offer a rich and diverse foundation of data, indispensable for the development and stringent validation of drug repositioning models.

Furthermore, we computed multimodal representations for drugs and diseases, informed by prior research [33]. For drugs, five types of similarities were calculated: chemical structure similarity  $\mathbf{R}_{cs}$  using SMILES and Tanimoto score [34]; ATC code similarity  $\mathbf{R}_{atc}$  based on therapeutic effects and chemical characteristics [35]; side effect similarity  $\mathbf{R}_{se}$  employing data from the SIDER database [36]; drug-drug interaction similarity  $\mathbf{R}_{ddi}$  derived from pharmacokinetics [29]; and target profile similarity  $\mathbf{R}_{dt}$  based on known drug targets [29]. For diseases, phenotype similarity  $\mathbf{C}_{ph}$  was assessed using phenotypic scores from the MimMiner database and disease ontology (DO) similarity  $\mathbf{C}_{do}$  based on gene ontology algorithms. We integrated these multimodal information sources by computing a weighted average of these similarity measures, described as follow:

$$\mathbf{R} = \frac{\mathbf{R}_{cs} + \mathbf{R}_{atc} + \mathbf{R}_{se} + \mathbf{R}_{ddi} + \mathbf{R}_{dt}}{5}. \quad (29)$$

$$\mathbf{C} = \frac{\mathbf{C}_{ph} + \mathbf{C}_{do}}{2}. \quad (30)$$

### F.1.2 TopoDR for drug repositioning

TopoDR comprises three main components: a multimodal data integration component, a network embedding component, and a prediction component (Figure 8). We use lighGCN+ as the network embedding component of the TopoDR framework. The prediction module utilizes both network embedding and multimodal data to depict drug-disease pairs, which can be defined as follows:

$$\text{Descriptor } (r, d) = [\mathbf{E}_r, \mathbf{E}_d, \mathbf{R}_r, \mathbf{C}_d] \quad (31)$$

We identify known associations as positive samples and correspondingly select an equal number of unacknowledged associations to serve as negative samples, thereby forming a training set. To predicting potential drug-disease associations, we employed a random forest algorithm as the classifier in this study.

## F.2 Experiments

### F.2.1 Performance evaluation

The 10-fold cross-validation was conducted to assess the performance of the drug repositioning model. In this process, all known drug-disease pairs were randomly divided into ten subsets, with each subset sequentially used as the test sample, while the remaining associations served as training samples. This cross-validation process was repeated ten times, and the averaged results were obtained. Additionally, six evaluation metrics were selected for assessing predictive performance: the area under the receiver operating characteristic curve (AUC), the area under the precision-recall curve (AUPR), accuracy (Acc.), precision (Pre.), recall (Rec.), and Matthew's correlation coefficient (MCC).

### F.2.2 Baseline model

To evaluate the performance of our proposed framework, we compared TopoDR with four existing drug repositioning methods listed below.

- deepDR [13] is a network-based deep learning model, integrating heterogeneous networks through a multi-modal deep autoencoder. It uses a variational autoencoder to encode and decode low-dimensional representations of drugs, identifying new uses for approved drugs.
- DRHGCN [14] is an approach using a graph convolutional network to integrate drug-drug, disease-disease similarities, and drug-disease associations. It combines intra-domain and inter-domain feature extraction with a layer attention mechanism for enhanced drug and disease embeddings.

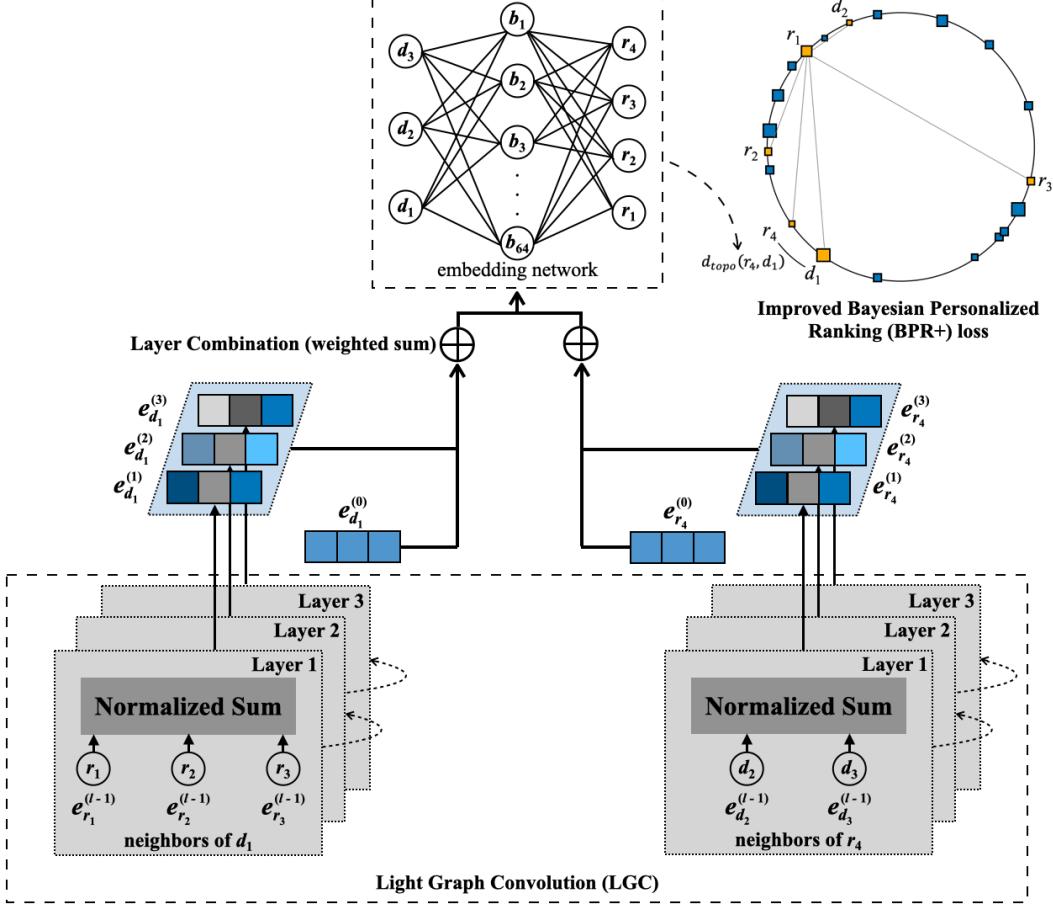


Figure 8: The flowchart of lightGCN+ model.

•DDAGDL[15] is a novel framework that integrates attention-based Graph Deep Learning with heterogeneous biomedical networks, enriched with biological knowledge. Utilizing non-Euclidean geometric priors and an attention mechanism, DDAGDL efficiently learns refined feature representations of drugs and diseases.

•AdaDR [16] is an adaptive GCN framework, leveraging an attention mechanism to fuse drug and disease features from topological and feature spaces for enhanced embeddings.

### F.2.3 The sensitivity analysis of parameters

In this section, we mainly examine the sensitivity analysis of parameters  $\lambda$  and  $\mu$  within 10 -fold cross-validation. TopoDR achieves its optimal performance when  $\lambda=1e-5$  and  $\mu = 1e-7$ . By altering one parameter while maintaining the other constant, we explore how such adjustments benefit the AUC values. It is noteworthy that we employed identical parameters for both the Fdataset and Cdataset.

Figure 9A depicts the performance trends of TopoDR with varying  $\lambda$  values. As  $\lambda$  varies, the AUC demonstrates a pattern of initial increase followed by a decrease. A downturn in AUC is observed when  $\lambda$  increases from  $1e-5$  to  $1e-7$ , indicating the necessity of assigning higher weight to global topological structure similarity information. Similarly, Figure 9B presents the performance trends of TopoDR across different  $\tau$  values, with the optimal AUC occurring at  $\tau=1e-7$ . A continuous decrease in AUC values is observed as  $\tau$  is reduced from  $1e-7$  to  $1e-9$ , suggesting that the regularization terms in Equation (7), specifically  $\|E^0\|^2$ , should not be excessively reduced.

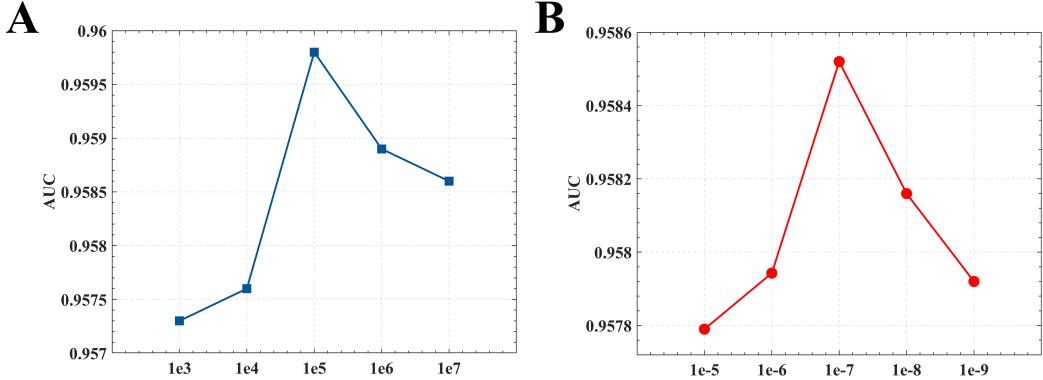


Figure 9: Effect of different  $\lambda$  and  $\tau$  on TopoDR performance. (A) Variation of the AUC values with the different settings of  $\lambda$ . (B) Variation of the AUC values with the different settings of  $\tau$ .

#### F.2.4 Ablation study

In this section, the efficacy of TopoDR is evaluated through comparative training across datasets with diverse feature types. Two scenarios are analyzed, each predicated on the utilization of distinct modalities within the proposed framework:

- TopoDR-w/o-N: AdaDR without network information.
- TopoDR-w/o-A: AdaDR without attribute information.

Table 4 reports the performance of each module in TopoDR. It clearly demonstrates that both modalities of TopoDR enhance model performance, with the network information providing considerable predictive accuracy. TopoDR-w/o-A shows notable precision, achieving 0.9677 in the Fdataset and 0.9804 in the Cdataset, playing a crucial role in minimizing false positives. TopoDR-w/o-N exhibits a balanced performance across various metrics. In the Fdataset, it achieves an AUC of 0.9397, accuracy of 0.8471, and recall of 0.7454, whereas in the Cdataset, it scores an AUC of 0.9444, accuracy of 0.8544, and recall of 0.7555. These results indicate its strong overall predictive capability, enhancing the model’s effectiveness in various aspects of prediction, from true positive rate to general accuracy and consistency. The synergistic interplay of these modalities enhances the model’s robustness and adaptability across various predictive scenarios.

Table 4: Performance of models associated with different modality across all datasets.

Dataset	Method	AUC	AUPR	Acc.	Pre.	Rec.	MCC
Fdataset	TopoDR-w/o-A	0.9312	0.9453	0.7998	<u>0.9677</u>	0.6239	0.6430
	TopoDR-w/o-N	<u>0.9397</u>	0.9455	<u>0.8471</u>	0.9412	<u>0.7454</u>	0.7120
	TopoDR	<b>0.9597</b>	<b>0.9581</b>	<b>0.8618</b>	<b>0.9683</b>	<b>0.7511</b>	<b>0.7440</b>
Cdataset	TopoDR-w/o-A	0.9377	<u>0.9529</u>	0.8323	<b>0.9804</b>	0.6793	0.6989
	TopoDR-w/o-N	<u>0.9444</u>	0.9511	<u>0.8544</u>	0.9431	<b>0.7555</b>	0.7240
	TopoDR	<b>0.9629</b>	<b>0.9686</b>	<b>0.8679</b>	<u>0.9753</u>	<b>0.7555</b>	<b>0.7556</b>

#### F.2.5 Comparison with graph self-supervised learning methods

To assess lightGCN+’s performance, we conducted comparisons with existing graph self-supervised learning methods (Node2vec [37], LINE [38], SDNE [39], and lightGCN [9]). As shown in Table 5, lightGCN+ consistently outperforms the other methods in almost all metrics, particularly in AUC and AUPR. These metrics are crucial as they represent the model’s ability to distinguish between classes effectively. In the case of the Fdataset, lightGCN+ shows superior performance, leading in AUC and AUPR, and displaying high accuracy and MCC. These results indicate not only its effectiveness in classification tasks but also its reliability in handling imbalanced datasets, as suggested by the high MCC values. For the Cdataset, lightGCN+ again stands out, particularly in AUC, AUPR, and Precision, suggesting its strong ability to predict positive classes accurately while reducing false

positives. The slight trade-off between Recall and Precision in favor of the latter is noticeable for lightGCN+, suggesting a tendency towards conservative classification strategies that prioritize precision. Comparatively, other methods like Node2vec, LINE, SDNE, and lightGCN show varying degrees of effectiveness. Node2vec and lightGCN, for instance, exhibit balanced performance across most metrics, suggesting their versatility. However, SDNE tends to lag slightly, particularly in the Fdataset, indicating possible limitations in capturing complex graph structures or generalizing across different datasets.

In summary, strengths of lightGCN+ in handling graph-structured data with a strong emphasis on precision and overall classification effectiveness. Its consistent performance across different datasets underscores its robustness and potential for broader applicability in graph-related machine learning tasks.

Table 5: The performance of existing graph self-supervised learning methods on Fdataset and Cdataset.

Dataset	Method	AUC	AUPR	Acc.	Pre.	Rec.	MCC
Fdataset	Node2vec	0.9483	0.9472	<u>0.8570</u>	<b>0.9698</b>	<u>0.7382</u>	<u>0.7360</u>
	LINE	0.9448	0.9399	0.8469	0.9514	0.7362	0.7144
	SDNE	0.9256	0.9254	0.8244	0.9508	0.6875	0.6764
	lightGCN	<u>0.9540</u>	<u>0.9523</u>	0.8495	0.9583	0.7336	0.7203
	lightGCN+	<b>0.9597</b>	<b>0.9581</b>	<b>0.8618</b>	<u>0.9683</u>	<b>0.7511</b>	<b>0.7440</b>
Cdataset	Node2vec	0.9537	0.9539	<u>0.8685</u>	0.9668	<u>0.7650</u>	<u>0.7546</u>
	LINE	0.9516	0.9588	<b>0.8687</b>	0.9601	<b>0.7713</b>	0.7529
	SDNE	0.9509	0.9542	0.8588	<u>0.9711</u>	0.7413	0.7395
	lightGCN	<u>0.9602</u>	<u>0.9592</u>	0.8640	0.9655	0.7584	0.7469
	lightGCN+	<b>0.9629</b>	<b>0.9686</b>	0.8679	<b>0.9753</b>	0.7555	<b>0.7556</b>

## F.2.6 Case studies

To verify the practical applicability of TopoDR in identifying potential therapeutic drugs for complex diseases, we conducted case studies. Using TopoDR on Fdataset, we predicted unknown drug-disease associations. For each disease, candidate drugs are ranked in descending order based on their predictive scores. In recent years, the development of anti-tumor drug has received increasing attention [40]. In this context, we selected four common cancers (colorectal cancer, breast cancer, gastric cancer, and leukemia) for case studies and retrieved evidence for candidate drugs from the CTD database [41]. Table 6 lists the top 10 candidate drugs for these cancers as ranked by the TopoDR, with the confirmed therapeutic drugs from the CTD database highlighted in bold. In the United States, approximately 12% of women are estimated to develop breast cancer over their lifetimes, a statistic underscored by the recording of over 250,000 new cases in 2017 alone [42]. Our drug candidates have undergone preliminary biological validation for breast cancer treatment. For instance, the study by Buxant et al. investigates the impact of dexamethasone (Dex) on MCF-7 breast cancer cells. Their findings indicate that Dex can inhibit cell proliferation, potentially via a pro-apoptotic mechanism, suggesting its therapeutic potential [43]. Similarly, the emergence of cisplatin as a treatment for metastatic triple-negative breast cancer (mTNBC) is attributable to its role as a DNA-damaging agent [44]. The rationale for using cisplatin in mTNBC is derived from the subtype’s unique molecular characteristics, notably its aberrant DNA repair mechanisms and extensive genomic instability. These features make mTNBC cells especially susceptible to DNA damage, thereby explaining the efficacy of cisplatin in this context. In summary, the outcomes of case studies indicate the practical significance of TopoDR in identifying potential therapeutic drugs for complex diseases.

Table 6: Top 10 candidate drugs for colorectal cancer, breast cancer, gastric cancer, and leukemia.

Diseases (OMIM IDs)	Top 10 candidate drugs (DrugBank IDs)
Colorectal Cancer (D114500)	<b>Doxorubicin (DB00997)</b> ; Teniposide (DB00444); <b>Prednisone (DB00635)</b> ; <b>Cisplatin (DB00515)</b> ; Bleomycin (DB00290); Busulfan (DB01008); Imatinib (DB00619); Conjugated Estrogens (DB00286); <b>Vincristine (DB00541)</b> ; Zoledronic Acid (DB00399).
Breast Cancer (D114480)	Bleomycin (DB00290); <b>Cisplatin (DB00515)</b> ; <b>Dacarbazine (DB00851)</b> ; Salmon Calcitonin (DB00017); Methylprednisolone (DB00959); Teniposide (DB00444); <b>Vincristine (DB00541)</b> ; Alendronic Acid (DB00630); Risedronic Acid (DB00884); Fludarabine (DB01073).
Gastric Cancer, Hereditary Diffuse (D137215)	Prednisone (DB00635); Vincristine (DB00541); Bleomycin (DB00290); Teniposide (DB00444); <b>Cisplatin (DB00515)</b> ; <b>Imatinib (DB00619)</b> ; Dacarbazine (DB00851); Zoledronic Acid (DB00399); Methylprednisolone (DB00959); Azathioprine (DB00993).
Breast Cancer (D109543)	<b>Dexamethasone (DB01234)</b> ; <b>Prednisone</b> ; <b>Cisplatin (DB00515)</b> ; <b>Doxorubicin (DB00997)</b> ; Triamcinolone (DB00620); <b>Daunorubicin (DB00694)</b> ; <b>Zoledronic Acid (DB00399)</b> ; Vinblastine (DB00570); Pamidronic Acid (DB00282); <b>Methylprednisolone (DB00959)</b> .

## NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer **[Yes]** , **[No]** , or **[NA]** .
- **[NA]** means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

**The checklist answers are an integral part of your paper submission.** They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "**[Yes]**" is generally preferable to "**[No]**", it is perfectly acceptable to answer "**[No]**" provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "**[No]**" or "**[NA]**" is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer **[Yes]** to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading “NeurIPS Paper Checklist”**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers.**

### 1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope?

Answer: **[Yes]**

Justification: The main claims made in the abstract and introduction accurately reflect the paper's contributions and scope.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

## 2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [\[Yes\]](#)

Justification: We discuss the limitations in the Conclusion section of the paper.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

## 3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [\[Yes\]](#)

Justification: For each theoretical result, we provide the full set of assumptions and a complete (and correct) proof.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.

- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

#### 4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [\[Yes\]](#)

Justification: We have provided detailed formulas and uploaded the corresponding code.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

#### 5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [\[Yes\]](#)

Justification: We have uploaded the code in the additional supplementary material.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.

- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

## 6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [\[Yes\]](#)

Justification: Experimental settings and details are discussed in detail in the appendix.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

## 7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [\[Yes\]](#)

Justification: The paper reports p-values for the main experiment, indicating that statistical significance tests have been performed.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.

- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

## 8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [\[Yes\]](#)

Justification: Experiments compute resources are discussed in detail in the appendix.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

## 9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [\[Yes\]](#)

Justification: The research conducted in the paper conforms in every respect with the NeurIPS Code of Ethics.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

## 10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [\[NA\]](#)

Justification: The research presented in this paper is foundational and does not directly tie to specific applications or deployments that could generate clear societal impacts, whether positive or negative. Given its theoretical and abstract nature, there is no direct pathway identified for either beneficial applications or potential misuse.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate

to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

## 11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: The paper does not involve the release of data or models that pose a high risk for misuse or require the implementation of specific safeguards for responsible release. The research focuses on theoretical aspects without deploying or providing access to potentially harmful models or datasets.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

## 12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The paper utilizes existing assets, including code and data sourced from GitHub. All creators or original owners of these assets are properly credited, and the licenses and terms of use are explicitly mentioned and respected. The paper includes citations to the original sources that produced the code packages or datasets, states the version of the assets used, and provides URLs where applicable. Each asset is accompanied by the name of the license under which it is released, ensuring compliance with the original terms of use.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, [paperswithcode.com/datasets](https://paperswithcode.com/datasets)

has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

### 13. New assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: The paper does not introduce or release any new assets.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

### 14. Crowdsourcing and research with human subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: The paper does not involve crowdsourcing experiments or research with human subjects, thereby eliminating the need for including instructions, screenshots, or details about compensation.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

### 15. Institutional review board (IRB) approvals or equivalent for research with human subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: The paper does not involve research with human subjects or crowdsourcing, thus eliminating the need for Institutional Review Board (IRB) approvals or discussions about potential risks and disclosures to participants.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

#### 16. Declaration of LLM usage

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The research did not involve the use of large language models (LLMs) in any part of the core method development. LLMs were only used for language editing and polishing, which did not influence the scientific rigor, originality, or methodology of the research.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.