SUPPLEMENTARY FILE FOR MGKAN

1. RELATED WORKS

1.1. Asymmetric DDI Prediction with GNNs

Most existing GNN-based DDI prediction models assume symmetric interactions and treat drug pairs as undirected edges in interaction networks [1, 2, 3, 4]. While effective, these models fail to distinguish source and target drug roles.

This limitation has prompted research into asymmetric DDI modeling. To capture directional pharmacological effects, asymmetric DDI prediction methods learn role-specific embeddings for source and target drugs. DGAT-DDI [5] is among the first to model asymmetric DDIs by introducing directed graph attention networks, which separately encode drugs as the source drug or the target drug based on asymmetric DDI networks. MAVGAE [6] adopts a variational graph autoencoder framework that integrates multimodal biochemical attributes (such as targets, enzymes, and pathways) into drug node features. DRGATAN [7] further models asymmetric DDIs as multi-relational directed edges and applies a relation-aware attention mechanism to capture directional semantics.

Despite leveraging multimodal features and directionality, these methods typically perform message passing solely within the DDI network, neglecting complementary information from higher-order relationships and biochemical properties. This limits their capacity to capture complex asymmetric patterns. Moreover, these methods predominantly rely on MLP-based GNN modules for message passing. Such modules may lack sufficient expressiveness to model heterogeneous, highly nonlinear interaction dynamics, motivating the exploration of more powerful functional representations.

1.2. Kolmogorov-Arnold Networks in Graph Learning

Kolmogorov-Arnold Networks (KANs), grounded in the Kolmogorov-Arnold representation theorem [8], have recently emerged as expressive alternatives to traditional MLPs. Rather than using fixed-weight edges and node activations, KANs employ learnable univariate basis functions along edges, enabling flexible and interpretable modeling of complex relationships [9]. KAN-based architectures have demonstrated strong performance across diverse domains, including physics-informed modeling, computer vision and forensics, and quantum computing [10]. Within graph learning, KAN modules have been integrated into GNNs for molecular property prediction (KA-GNN [11], GNN-SKAN [12]), recommendation systems (FourierKAN-GCF [13]), and social network analysis (KAGNNS [14], GKAN [15]). These studies highlight KANs' capacity to enhance GNNs via basis-function-driven transformations.

However, integrating KANs into multimodal, directed graph architectures for asymmetric DDI prediction remains unexplored. This gap motivates our work, where we propose a KAN-augmented framework designed to capture both heterogeneous modalities and directional semantics in asymmetric DDI networks.

2. THEORETICAL ANALYSIS OF SECOND-ORDER SIMILARITY KERNELS

The spectral properties of second-order similarity kernels are analyzed to understand their propagation behavior. We rigorously justify why the second-order similarity kernels C_{in} and C_{out} yield propagation dynamics that avoid the homogenization typical of Laplacian- or random-walk-based convolutions.

Let $\mathcal{G} = (V, E)$ be a strongly connected digraph with adjacency matrix $A \in \mathbb{R}^{n\times n}$. Define:

$$C_{in}(i,j) = \sum_k \frac{A_{k,i}A_{k,j}}{\sum_v A_{k,v}}, C_{out}(i,j) = \sum_k \frac{A_{i,k}A_{j,k}}{\sum_v A_{v,k}}.$$

Each C_{in}, C_{out} captures degree-normalized two-hop similarity: C_{in} encodes shared predecessors (co-accessibility), while C_{out} encodes shared successors (co-influence). Let their degree matrices be

$$D_{C_{in}} = \mathrm{diag}\Big(\sum_j C_{in}(i,j)\Big), D_{C_{out}} = \mathrm{diag}\Big(\sum_j C_{out}(i,j)\Big).$$

Define the normalized operator:

$$\hat{C}_{in} = D_{C_{in}}^{-1/2} C_{in} D_{C_{in}}^{-1/2}.$$

Proposition 1 (Spectral Structure and Convergence)**.** *For a strongly connected $\mathcal{G}$, the normalized kernel $\hat{C}_{in}$ satisfies:*

1. *Positive Semidefiniteness. C_{in} admits a Gram decomposition:*

$$C_{in} = \Phi^\top \Phi, \qquad \Phi_{k,j} = \frac{A_{k,j}}{\sqrt{\sum_v A_{k,v}}}.$$

Hence C_{in} is symmetric PSD, and $\operatorname{rank}(C_{in}) \leq \operatorname{rank}(A)$.

2. *Perron Eigenpair. $\hat{C}_{in}$ is real symmetric, with eigenvalues*

$$1 = \lambda_1 > \lambda_2 \geq \cdots \geq \lambda_n \geq 0.$$

The top eigenvector is explicitly

$$u_1 = \frac{D_{C_{in}}^{1/2}\mathbf{1}}{\|D_{C_{in}}^{1/2}\mathbf{1}\|_2}, \qquad \hat{C}_{in}u_1 = u_1,$$

and all other eigenvalues satisfy $|\lambda_i| < 1$.

3. *Convergence of Propagation. For any $X \in \mathbb{R}^{n\times d}$, iterating*

$$H^{(k+1)} = \hat{C}_{in}H^{(k)}, \qquad H^{(0)} = X,$$

yields

$$H^{(k)} = \sum_{i=1}^{n} \lambda_i^k u_i u_i^\top X.$$

As $k \to \infty$, all terms with $i \geq 2$ vanish, and

$$H^{(\infty)} = u_1 u_1^\top X.$$

4. *Non-Uniform Steady State. Since*

$$[D_{C_{in}}]_{ii} = \sum_j C_{in}(i,j) = \sum_k \frac{A_{k,i}}{\sum_v A_{k,v}} \sum_j A_{k,j},$$

we have

$$u_1(i) \propto \sqrt{\sum_k \frac{A_{k,i}}{\sum_v A_{k,v}} \sum_j A_{k,j}}.$$

Unless $\mathcal{G}$ is regular, u_1 is non-uniform, and the limit embeddings maintain nonzero variance:

$$\mathcal{V}(H^{(\infty)}) = \frac{1}{n}\sum_{i=1}^{n} \|H_i^{(\infty)} - \bar{H}^{(\infty)}\|_2^2 > 0.$$

Proof. (1) follows by directly factorizing each term of C_{in} as a rank-one PSD matrix:

$$C_{in} = \sum_k \frac{1}{\sum_v A_{k,v}} (A_{k,:})^\top (A_{k,:}),$$

with Φ defined as above.

(2) Since C_{in} is symmetric, $\hat{C}_{in}$ is real symmetric. Moreover,

$$\hat{C}_{in} D_{C_{in}}^{1/2}\mathbf{1} = D_{C_{in}}^{-1/2} C_{in}\mathbf{1} = D_{C_{in}}^{1/2}\mathbf{1},$$

so $u_1 \propto D_{C_{in}}^{1/2}\mathbf{1}$ is a normalized eigenvector with eigenvalue 1. Strong connectivity implies irreducibility, hence by Perron–Frobenius all other eigenvalues satisfy $|\lambda_i| < 1$.

(3) The eigendecomposition $\hat{C}_{in} = U\Lambda U^\top$ yields the stated expansion. Since $|\lambda_i| < 1$ for $i \geq 2$, only the $i = 1$ term survives as $k \to \infty$.

(4) Substituting the explicit form of $D_{C_{in}}$ gives the stated expression for u_1. Non-regularity implies non-uniform u_1, hence the node-wise embeddings $H^{(\infty)}$ retain structural variance.

□

Remark 1. *This result shows that second-order kernels mitigate over-smoothing: the steady state is not a uniform consensus (as for Laplacians) nor a trivial stationary mixture (as for first-order random walks), but a degree-structured projection that preserves meaningful node-level variation.*

3. DATASETS AND SETUP

3.1. Datasets.

To validate the effectiveness of MGKAN, we conduct experiments on two datasets of different scales, DS 1 and DS 2. These datasets are constructed from different versions of DrugBank [16].

- **DS 1**: Based on DrugBank v5.1.2, we filtered out symmetric interactions and removed drugs with invalid SMILES strings, resulting in 1,659 drugs and 129,803 asymmetric DDIs.
- **DS 2**: Based on DrugBank v5.1.12, we applied the same filtering procedure, yielding 2,124 drugs and 428,949 asymmetric DDIs.

In addition, we also collected the targets, enzymes, and transporters associated with each drug from DrugBank, as shown in Table 1.

Table 1. The statistics of two datasets.

Dataset	Drugs	DDIs	Targets	Enzymes	Transporters
DS 1	1,659	129,803	1,571	323	205
DS 2	2,124	428,949	2,093	386	272

3.2. Baselines.

We compare MGKAN with seven state-of-the-art baselines, including four general-purpose directed graph learning methods—Gravity VGAE and S/T VGAE [17], DiGAE [18], and DirGNN [19]—as well as three models specifically designed for asymmetric DDI prediction: DGAT-DDI [5], MAVGAE [6], and DRGATAN [7]. These baselines reflect both the structural properties of directed graphs and task-specific asymmetry modeling for DDIs.

3.3. Evaluation metrics.

The performance of our model and baselines was evaluated using four widely adopted metrics: Area Under the ROC Curve (AUROC), Area Under the P-R Curve (AUPRC), Accuracy (ACC), and F1-score.

3.4. Experimental setup.

We reduce the 1024-dimensional Morgan fingerprints to 100-dimensional vectors via PCA, which are used as initial node features. All models are evaluated using five-fold cross-validation. In each fold, known DDIs are randomly split into training, validation, and test sets. Negative samples are sampled from unknown DDIs without overlap with positive samples and are matched in number to the positive samples in each split.

For the experimental settings, we set the epoch to 200, the learning rate to 0.01, and use the Adam optimizer. Furthermore, the embedding dimension is set to $d = 32$, the number of GKAN hidden layers is $l = 1$, the spline order of KAN is $k = 3$, and the grid size is $g = 15$. All baselines are retrained using their original hyperparameters.

4. PERFORMANCE COMPARISON

We evaluate MGKAN on the specific-direction (Task 1) and asymmetric-direction (Task 2) prediction tasks on the DS 1 and DS 2 datasets, as reported in Tables 2 and 3. To assess efficiency, we include a runtime efficiency analysis in **Supplementary Material**, which shows that MGKAN maintains competitive runtime performance.

4.1. Task 1: specific-direction prediction.

As shown in Table 2, MGKAN consistently outperforms all baselines on Task 1. While directed graph learning methods exhibit strong structural modeling capabilities, they achieve lower AUROC and F1, indicating that structural information alone is insufficient for accurate asymmetric DDI prediction. On the other hand, existing asymmetric DDI baselines primarily rely on the DDI network, limiting their ability to capture complex interaction patterns. MGKAN addresses these limitations by integrating multiple views—including co-interaction and similarity networks—which provide complementary topological and semantic information. For instance, drugs without direct interactions can still be indirectly connected through co-interaction networks, while the similarity view introduces auxiliary biochemical features. This multi-view architecture enables MGKAN to generate more expressive and discriminative embeddings than models restricted to a single view.

4.2. Task 2: asymmetric-direction prediction.

In contrast, Task 2 is more challenging, as it requires precise identification of the interaction direction. As shown in Table 3, MGKAN achieves the best performance on both datasets, improving accuracy by 2.02% on DS 1 and 3.85% on DS 2 over the strongest baseline, DGAT-DDI. While baselines such as DGAT-DDI and DRGATAN incorporate directional message passing, MGKAN enhances directional modeling by integrating KANs into both the message passing and fusion stages. Compared to conventional MLPs used in other GNN baselines, the learnable basis functions in KANs allow MGKAN to capture nonlinear directional patterns more effectively. In addition, its complementary fusion module integrates multi-view embeddings, further enriching the learned representations. These designs collectively contribute to MGKAN's superior performance in asymmetric DDI prediction.

5. VISUALIZATION OF ROLE-SPECIFIC EMBEDDINGS

To evaluate the capability of MGKAN for learning directed role-specific embeddings, we randomly selected 500 drugs from the dataset, extracted their corresponding source and target embeddings, and applied t-SNE for visualization. The embedding distributions learned by a standard GCN and MGKAN are visualized in Figure 1 for comparison.

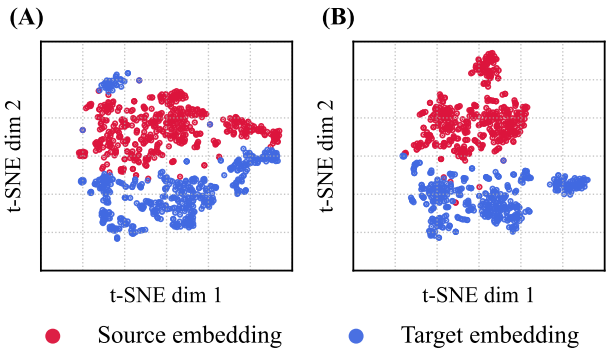

Fig. 1. The t-SNE visualization of role-specific embeddings. **(A)** Role-specific embeddings learned by GCN. **(B)** Role-specific embeddings learned by MGKAN.

The visualization results show that while the embeddings learned by standard GCN exhibit a degree of clustering, there is substantial overlap between source and target roles, indicating limited capacity for role differentiation. In contrast, MGKAN produces more clearly clustered and well-separated embeddings in both directions, reflecting a stronger awareness of directional semantics. These findings validate the effectiveness of MGKAN's direction-aware design and the GKAN-based encoding architecture in capturing asymmetric role-specific embeddings.

Table 2. Performance comparison of MGKAN in **Task 1**. Results are reported as the mean and standard deviation (%) over five-fold cross-validation. The highest value in each column is highlighted in bold.

Method	DS 1				DS 2			
	AUROC	AUPRC	ACC	F1	AUROC	AUPRC	ACC	F1
Gravity VGAE	$88.54_{\pm 0.01}$	$85.26_{\pm 0.03}$	$81.66_{\pm 0.03}$	$82.20_{\pm 0.03}$	$87.90_{\pm 0.01}$	$85.75_{\pm 0.01}$	$79.93_{\pm 0.02}$	$80.57_{\pm 0.03}$
S/T VGAE	$92.14_{\pm 0.08}$	$91.51_{\pm 0.10}$	$84.13_{\pm 0.13}$	$83.72_{\pm 0.22}$	$91.26_{\pm 0.04}$	$90.75_{\pm 0.07}$	$83.23_{\pm 0.12}$	$82.94_{\pm 0.26}$
DiGAE	$94.84_{\pm 0.03}$	$93.73_{\pm 0.06}$	$91.15_{\pm 0.04}$	$91.29_{\pm 0.03}$	$90.57_{\pm 0.02}$	$88.44_{\pm 0.05}$	$85.91_{\pm 0.03}$	$86.14_{\pm 0.02}$
DirGNN	$97.82_{\pm 0.02}$	$97.38_{\pm 0.04}$	$93.24_{\pm 0.03}$	$93.22_{\pm 0.03}$	$95.55_{\pm 0.02}$	$94.99_{\pm 0.03}$	$89.18_{\pm 0.03}$	$89.13_{\pm 0.04}$
MAVGAE	$92.53_{\pm 0.06}$	$91.78_{\pm 0.06}$	$84.39_{\pm 0.07}$	$84.21_{\pm 0.06}$	$91.57_{\pm 0.08}$	$91.57_{\pm 0.08}$	$83.29_{\pm 0.44}$	$82.81_{\pm 0.77}$
DRGATAN	$87.30_{\pm 0.05}$	$86.72_{\pm 0.05}$	$79.82_{\pm 0.08}$	$80.71_{\pm 0.08}$	$83.51_{\pm 0.05}$	$83.88_{\pm 0.05}$	$74.48_{\pm 0.06}$	$76.85_{\pm 0.05}$
DGAT-DDI	$98.55_{\pm 0.03}$	$98.20_{\pm 0.04}$	$94.69_{\pm 0.12}$	$94.67_{\pm 0.14}$	$96.65_{\pm 0.02}$	$95.98_{\pm 0.03}$	$90.91_{\pm 0.04}$	$91.02_{\pm 0.06}$
MGKAN	$\mathbf{99.08_{\pm 0.01}}$	$\mathbf{98.94_{\pm 0.02}}$	$\mathbf{95.05_{\pm 0.14}}$	$\mathbf{94.90_{\pm 0.16}}$	$\mathbf{98.09_{\pm 0.02}}$	$\mathbf{97.62_{\pm 0.03}}$	$\mathbf{93.04_{\pm 0.09}}$	$\mathbf{92.93_{\pm 0.11}}$

Table 3. Performance comparison of MGKAN in **Task 2**. Results are reported as the mean and standard deviation (%) over five-fold cross-validation. The highest value in each column is shown in bold.

Method	DS 1				DS 2			
	AUROC	AUPRC	ACC	F1	AUROC	AUPRC	ACC	F1
Gravity VGAE	$69.48_{\pm 0.02}$	$62.33_{\pm 0.06}$	$65.80_{\pm 0.03}$	$71.28_{\pm 0.04}$	$66.33_{\pm 0.02}$	$61.87_{\pm 0.03}$	$61.92_{\pm 0.02}$	$61.92_{\pm 0.02}$
S/T VGAE	$85.05_{\pm 0.21}$	$72.28_{\pm 0.41}$	$69.19_{\pm 0.36}$	$68.33_{\pm 0.21}$	$83.03_{\pm 0.04}$	$83.03_{\pm 0.04}$	$66.49_{\pm 0.11}$	$65.45_{\pm 0.04}$
DiGAE	$90.91_{\pm 0.04}$	$82.12_{\pm 0.09}$	$66.21_{\pm 0.09}$	$66.59_{\pm 0.09}$	$83.48_{\pm 0.02}$	$69.44_{\pm 0.03}$	$58.53_{\pm 0.04}$	$58.53_{\pm 0.04}$
DirGNN	$94.62_{\pm 0.01}$	$88.14_{\pm 0.02}$	$85.31_{\pm 0.05}$	$84.79_{\pm 0.05}$	$89.96_{\pm 0.02}$	$79.37_{\pm 0.05}$	$76.91_{\pm 0.04}$	$76.29_{\pm 0.03}$
MAVGAE	$88.51_{\pm 0.15}$	$77.92_{\pm 0.26}$	$73.50_{\pm 0.26}$	$72.65_{\pm 0.24}$	$85.96_{\pm 0.14}$	$73.52_{\pm 0.22}$	$70.11_{\pm 0.22}$	$69.08_{\pm 0.22}$
DRGATAN	$83.43_{\pm 0.10}$	$70.61_{\pm 0.11}$	$65.42_{\pm 0.07}$	$65.31_{\pm 0.07}$	$78.15_{\pm 0.04}$	$61.62_{\pm 0.05}$	$54.78_{\pm 0.09}$	$54.98_{\pm 0.11}$
DGAT-DDI	$95.50_{\pm 0.06}$	$89.79_{\pm 0.12}$	$87.30_{\pm 0.12}$	$86.86_{\pm 0.10}$	$90.94_{\pm 0.06}$	$80.88_{\pm 0.09}$	$78.18_{\pm 0.11}$	$77.62_{\pm 0.09}$
MGKAN	$\mathbf{96.21_{\pm 0.02}}$	$\mathbf{91.07_{\pm 0.03}}$	$\mathbf{89.32_{\pm 0.10}}$	$\mathbf{88.87_{\pm 0.10}}$	$\mathbf{92.83_{\pm 0.03}}$	$\mathbf{84.02_{\pm 0.05}}$	$\mathbf{82.03_{\pm 0.16}}$	$\mathbf{81.53_{\pm 0.18}}$

6. RUNTIME EFFICIENCY ANALYSIS

To assess runtime efficiency, we measured the average training time per fold (five-fold cross-validation) of each model on Task 1 – DS 2 using a single NVIDIA RTX 4090 GPU.

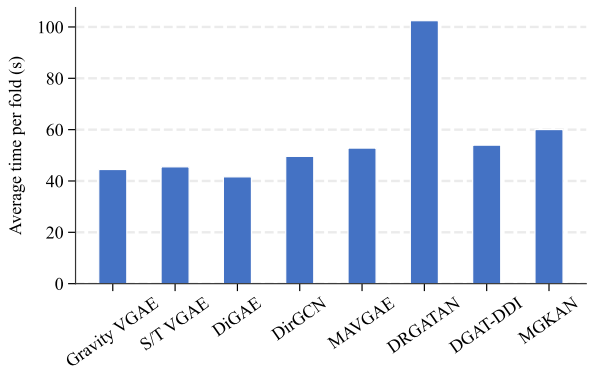

Fig. 2. Per-fold runtime of all models (s)

Figure 2 presents the runtime results. DiGAE is the fastest baseline at 41.6 s, followed by Gravity VGAE (44.47s), S/T VGAE (45.51 s), DirGCN (49.58 s), MAVGAE (52.8 1s), and the best-performing baseline DGAT-DDI (53.95 s); DRGATAN is the slowest at 102.44 s. MGKAN requires 60.03 s, incurring an overhead of +18.43 s (+44%) compared to the fastest model and +6.08 s (+11%) over DGAT-DDI, yet it consistently achieves the best performance across all metrics, tasks, and datasets. These results demonstrate that MGKAN delivers state-of-the-art accuracy while maintaining a modest runtime overhead, offering a favorable accuracy–efficiency trade-off for asymmetric DDI prediction.

7. REFERENCES

[1] Marinka Zitnik, Monica Agrawal, and Jure Leskovec, "Modeling polypharmacy side effects with graph convolutional networks," *Bioinformatics*, vol. 34, no. 13, pp. i457–i466, 2018.

[2] Yue-Hua Feng, Shao-Wu Zhang, and Jian-Yu Shi, "Dpddi: a deep predictor for drug-drug interactions," *BMC bioinformatics*, vol. 21, no. 1, pp. 419, 2020.

[3] Yujie Chen, Tengfei Ma, Xixi Yang, Jianmin Wang, Bosheng Song, and Xiangxiang Zeng, "Muffin: multi-

scale feature fusion for drug–drug interaction prediction," *Bioinformatics*, vol. 37, no. 17, pp. 2651–2658, 2021.

[4] Jing Wang, Shuo Zhang, Runzhi Li, Gang Chen, Siyu Yan, and Lihong Ma, "Multi-view feature representation and fusion for drug-drug interactions prediction," *BMC bioinformatics*, vol. 24, no. 1, pp. 93, 2023.

[5] Yi-Yang Feng, Hui Yu, Yue-Hua Feng, and Jian-Yu Shi, "Directed graph attention networks for predicting asymmetric drug–drug interactions," *Briefings in Bioinformatics*, vol. 23, no. 3, pp. bbac151, 2022.

[6] Zengqian Deng, Jie Xu, Yinfei Feng, Liangcheng Dong, and Yuanyuan Zhang, "Mavgae: a multimodal framework for predicting asymmetric drug–drug interactions based on variational graph autoencoder," *Computer Methods in Biomechanics and Biomedical Engineering*, pp. 1–13, 2024.

[7] Dehai Zhang, Zhengwu Wang, Di Zhao, and Jin Li, "Drgatan: Directed relation graph attention aware network for asymmetric drug-drug interaction prediction," *Iscience*, vol. 27, no. 6, 2024.

[8] AN Kolmogorov, "On the representation of continuous functions of many variables by superposition of continuous functions of one variable and addition," *Doklady Akademii Nauk*, vol. 114, pp. 953–956, 1957.

[9] Ziming Liu, Yixuan Wang, Sachin Vaidya, Fabian Ruehle, James Halverson, Marin Soljačić, Thomas Y Hou, and Max Tegmark, "Kan: Kolmogorov-arnold networks," *arXiv preprint arXiv:2404.19756*, 2024.

[10] Shriyank Somvanshi, Syed Aaqib Javed, Md Monzurul Islam, Diwas Pandit, and Subasish Das, "A survey on kolmogorov-arnold network," *ACM Computing Surveys*, 2024.

[11] Longlong Li, Yipeng Zhang, Guanghui Wang, and Kelin Xia, "Ka-gnn: Kolmogorov-arnold graph neural networks for molecular property prediction," *arXiv preprint arXiv:2410.11323*, 2024.

[12] Ruifeng Li, Mingqian Li, Wei Liu, and Hongyang Chen, "Gnn-skan: harnessing the power of swallowkan to advance molecular representation learning with gnns," *arXiv preprint arXiv:2408.01018*, 2024.

[13] Jinfeng Xu, Zheyu Chen, Jinze Li, Shuo Yang, Wei Wang, Xiping Hu, and Edith C-H Ngai, "Fourierkan-gcf: Fourier kolmogorov-arnold network–an effective and efficient feature transformation for graph collaborative filtering," *arXiv preprint arXiv:2406.01034*, 2024.

[14] Roman Bresson, Giannis Nikolentzos, George Panagopoulos, Michail Chatzianastasis, Jun Pang, and Michalis Vazirgiannis, "Kagnns: Kolmogorov-arnold networks meet graph learning," *arXiv preprint arXiv:2406.18380*, 2024.

[15] Mehrdad Kiamari, Mohammad Kiamari, and Bhaskar Krishnamachari, "Gkan: Graph kolmogorov-arnold networks," *arXiv preprint arXiv:2406.06470*, 2024.

[16] Craig Knox, Mike Wilson, Christen M Klinger, Mark Franklin, Eponine Oler, Alex Wilson, Allison Pon, Jordan Cox, Na Eun Chin, Seth A Strawbridge, et al., "Drugbank 6.0: the drugbank knowledgebase for 2024," *Nucleic acids research*, vol. 52, no. D1, pp. D1265–D1275, 2024.

[17] Guillaume Salha, Stratis Limnios, Romain Hennequin, Viet-Anh Tran, and Michalis Vazirgiannis, "Gravity-inspired graph autoencoders for directed link prediction," in *Proceedings of the 28th ACM international conference on information and knowledge management*, 2019, pp. 589–598.

[18] Georgios Kollias, Vasileios Kalantzis, Tsuyoshi Idé, Aurélie Lozano, and Naoki Abe, "Directed graph autoencoders," in *Proceedings of the AAAI conference on artificial intelligence*, 2022, vol. 36, pp. 7211–7219.

[19] Emanuele Rossi, Bertrand Charpentier, Francesco Di Giovanni, Fabrizio Frasca, Stephan Günnemann, and Michael M Bronstein, "Edge directionality improves learning on heterophilic graphs," in *Learning on graphs conference*. PMLR, 2024, pp. 25–1.