nature computational science

Article

https://doi.org/10.1038/s43588-023-00558-4

Emerging drug interaction prediction enabled by a flow-based graph neural network with biomedical network

Received: 6 June 2023

Accepted: 25 October 2023

Published online: 20 December 2023

Check for updates

Yongqi Zhang[1], Quanming Yao[2] ✉, Ling Yue[2], Xian Wu[3], Ziheng Zhang[3], Zhenxi Lin[3] & Yefeng Zheng[3]

Drug–drug interactions (DDIs) for emerging drugs offer possibilities for treating and alleviating diseases, and accurately predicting these with computational methods can improve patient care and contribute to efficient drug development. However, many existing computational methods require large amounts of known DDI information, which is scarce for emerging drugs. Here we propose EmerGNN, a graph neural network that can effectively predict interactions for emerging drugs by leveraging the rich information in biomedical networks. EmerGNN learns pairwise representations of drugs by extracting the paths between drug pairs, propagating information from one drug to the other, and incorporating the relevant biomedical concepts on the paths. The edges of the biomedical network are weighted to indicate the relevance for the target DDI prediction. Overall, EmerGNN has higher accuracy than existing approaches in predicting interactions for emerging drugs and can identify the most relevant information on the biomedical network.

Scientific advances and regulatory changes have led to the development of numerous emerging drugs worldwide, in particular for rare, severe or life-threatening illnesses[1,2]. These drugs are novel substances with unknown or unpredictable risks, as they have not been extensively regulated or used before. For example, although hundreds of COVID-19 drugs have been developed, only six have been recommended by the Food and Drug Administration as of October 2023 (for example, dexamethasone and hydrocortisone). Clinical deployment of new drugs is cautious and slow, making it crucial to identify drug–drug interactions (DDIs) for these emerging drugs. To speed up the discovery of potential DDIs, computational techniques, particularly machine learning approaches, have been developed[3–6]. However, with limited clinical trial information, unexpected polypharmacy or side effects can be severe and difficult to detect[7,8].

Early DDI prediction methods used fingerprints[9] or hand-designed features[4,10] to indicate interactions based on drug properties. Although these methods can work directly on emerging drugs in a cold-start setting[10,11], they can lack expressiveness and ignore the mutual information between drugs. DDI facts can naturally be represented as a graph where nodes represent drugs and edges represent interactions between a pair of drugs. Graph learning methods can learn drug embeddings for prediction[12], but they rely on historical interactions and thus struggle with the problem of scarce interaction data for emerging drugs.

Incorporating large biomedical networks as side information for DDI prediction is an alternative to learning solely from DDI interactions[5,6,13–17]. These biomedical networks, such as HetioNet[18], organize facts into a directed multi-relational graph, recording relationships between biomedical concepts, such as genes, diseases and drugs. Tanvir and colleagues used hand-designed meta-paths from a biomedical network[5], while Karim and colleagues learned embeddings from the network and used a deep network to perform DDI prediction[14]. Graph neural networks[19,20] can obtain expressive node embeddings by aggregating topological structure and drug embeddings, but the

[1]4Paradigm Inc., Beijing, China. [2]Department of Electronic Engineering, Tsinghua University, Beijing, China. [3]Tencent Jarvis Lab, Shenzhen, China. ✉e-mail: qyaoaa@tsinghua.edu.cn

https://doi.org/10.1038/s43588-023-00558-4

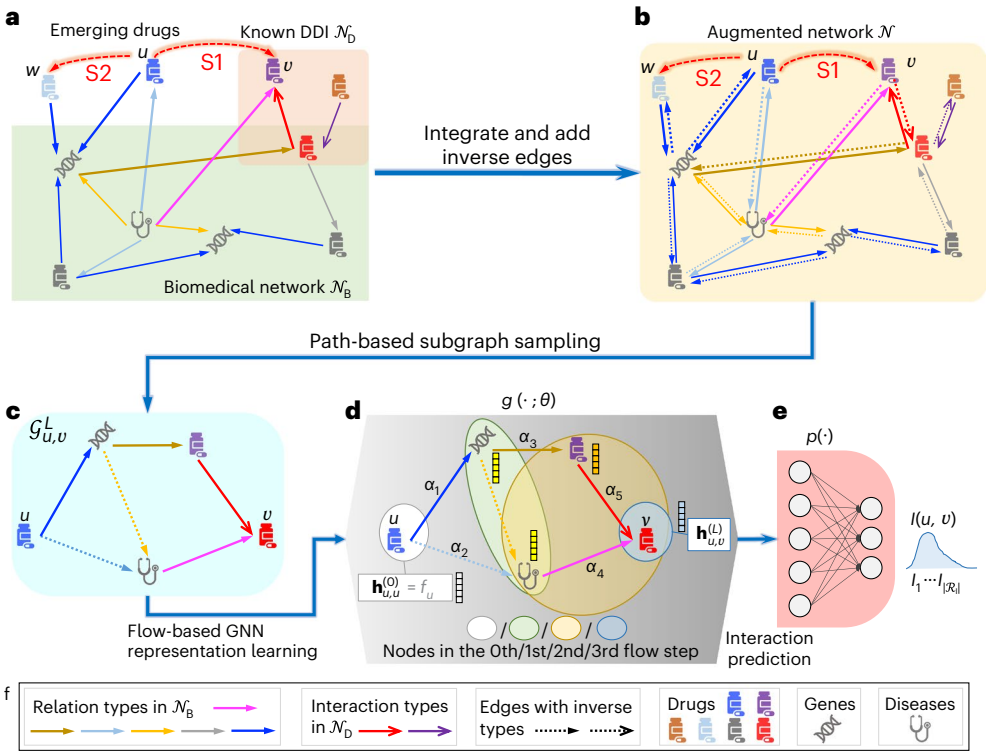

Fig. 1 | Overview of EmerGNN. a, Problem formulation: given a DDI network of existing drugs $\mathcal{N}_D$ and a large biomedical network $\mathcal{N}_B$ that provides side information for the drugs, the task is to predict the interaction type between an emerging drug (such as u, in dark blue) and an existing drug (such as v, in purple) in the S1 setting, or the interaction type between two emerging drugs (such as u in dark blue and w in light blue) in the S2 setting. **b**, Augmented network $\mathcal{N}$. The DDI network and biomedical network are integrated, and edges with inverse types are incorporated to obtain an augmented network $\mathcal{N}$. The augmentation brings better communication among drugs and entities in both the interaction and biomedical networks. **c**, Path-based subgraph. Given a drug pair (u, v) to be predicted, all the paths from u to v with length no larger than L are extracted to construct a path-based subgraph $\mathcal{G}^L_{u,v}$. **d**, Flow-based GNN $g(\cdot;\theta)$ with parameters θ. The network flows the initial drug features $\mathbf{h}^{(0)}_{u,u} = \mathbf{f}_u$ over essential information in $\mathcal{G}^L_{u,v}$ for L steps. It uses different attention weights α' to weight the importance of the different edges. After L steps, a pairwise representation $\mathbf{h}^{(L)}_{u,v}$ between u and v is obtained as the subgraph representation of $\mathcal{G}^L_{u,v}$. **e**, Interaction predictor $p(\cdot)$. A simple linear classifier $p(\mathbf{h}^{(L)}_{u,v})$ outputs a distribution $\boldsymbol{I}(u, v)$, where each dimension indicates an interaction type $i \in \mathcal{R}_I$ between u and v. **f**, The different relation and interaction types are indicated by arrows with different colors. Edges with inverse types are indicated by dashed arrows of the corresponding color. The icons represent biomedical concepts: drugs, genes and diseases.

existing methods[6,13,15–17] do not specially consider emerging drugs, leading to poor performance when predicting DDIs for them.

In this Article we propose to use a large biomedical network to predict DDIs for emerging drugs by learning from the biomedical concepts connecting target drug pairs. Although emerging drugs may not have sufficient interactions in the DDI network, they often share the same biochemical concepts as used in the drug development of existing drugs, such as targeted genes or diseases. We thus exploit related paths from the biomedical networks for given drug pairs. However, properly utilizing these networks can be challenging, as they were not developed for emerging drugs, and the mismatch of objectives can lead machine learning models to learn distracting knowledge.

To accurately and interpretably predict DDIs for emerging drugs, here we introduce EmerGNN, a graph neural network (GNN) method that learns pairwise drug representations by integrating the biomedical entities and relations connecting them. A flow-based GNN architecture extracts paths connecting drug pairs, traces from an emerging drug to an existing drug, and integrates information about the biomedical concepts along the paths. This approach utilizes shared information in the biomedical and interaction networks. To enable the extraction of relevant information, we weight different types of relation on the biomedical network, such that edges with larger weights on the paths are helpful for interpretation. Compared with other GNN-based methods, EmerGNN propagates on the local subgraph around the drug pair to be predicted, and better discovers directional information flow within the biomedical network. In summary, our main contributions are as follows:

- Building on a biomedical network, we develop an effective deep learning method that accurately predicts interactions for emerging drugs.
- We propose EmerGNN, a GNN-based method that learns pairwise representations of drug pairs to predict DDIs for emerging drugs by integrating the relevant biomedical concepts connecting them.
- Extensive experiments show that EmerGNN is effective in predicting interactions for emerging drugs. The concepts learned on the biomedical network are interpretable.
- EmerGNN's strong prediction ability has the potential to clinically improve patient care and contribute to more efficient drug development processes.

Results

EmerGNN, encoding pairwise representations with flow-based GNN for emerging drugs

Here we focus on two DDI prediction task settings for emerging drugs [10,11,21] (Fig. 1a and Methods): the S1 setting, determining the interaction type between an emerging drug and an existing drug, and the S2 setting, determining the interaction type between two emerging drugs. To connect emerging and existing drugs, we use a large biomedical network, HetioNet[18], which contains entities and relations related to

https://doi.org/10.1038/s43588-023-00558-4

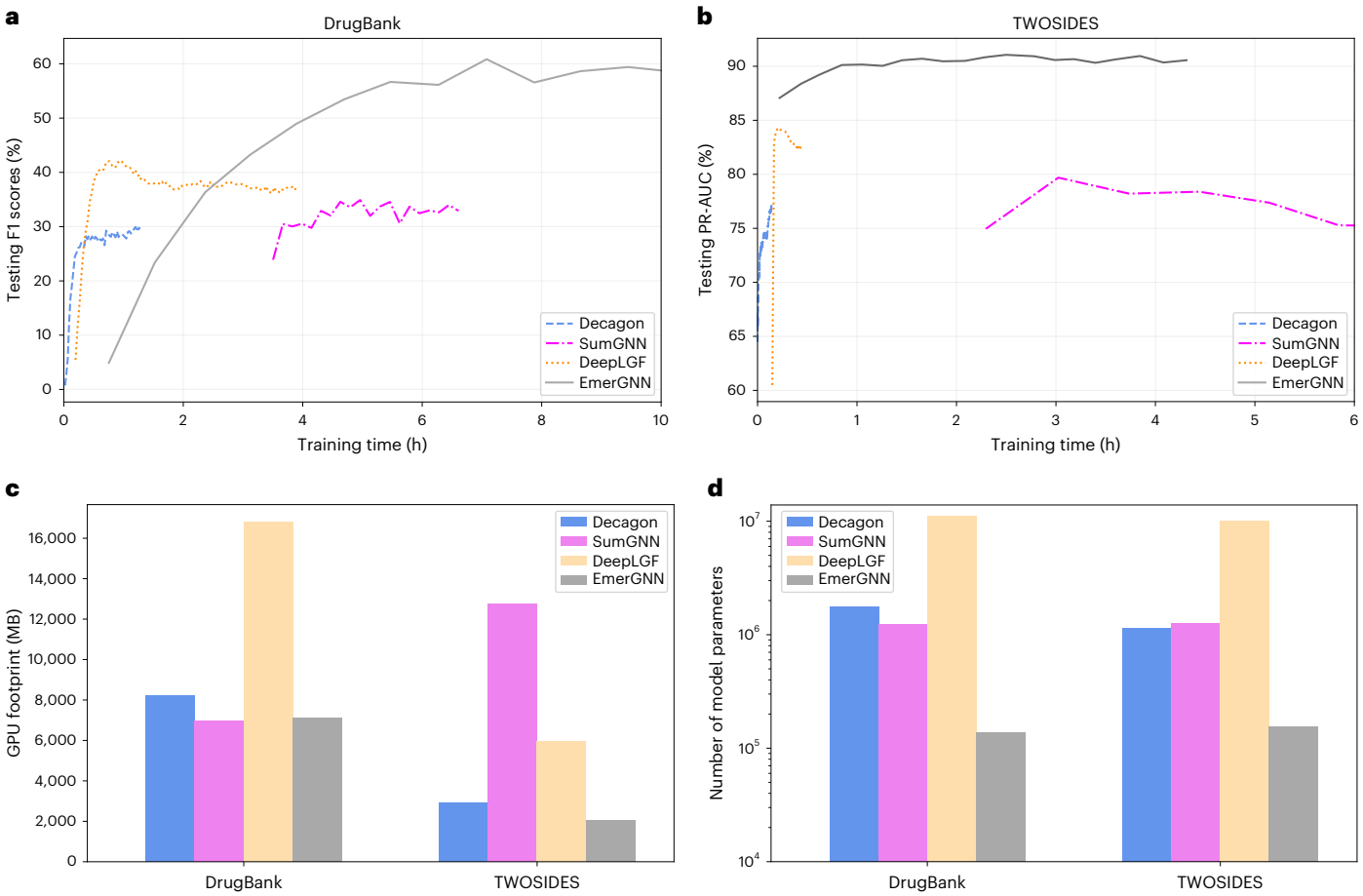

Fig. 2 | Complexity analysis of different GNN-based methods in the S1 setting. a, Comparison of training curves on the DrugBank dataset. **b**, Comparison of training curves on the TWOSIDES dataset. **c**, Comparison of GPU memory footprint usage on the two datasets in megabytes. **d**, Comparison of the number of trainable model parameters on the two datasets.

biomedical concepts. We assume that all the emerging drugs are connected to entities in the biomedical network, allowing us to infer their properties from existing drugs and the biomedical network.

Given a DDI network and a biomedical network (Fig. 1a), we first integrate the two networks to enable communication between existing and emerging drugs that are connected by biomedical concepts, such as proteins, diseases or other drugs, and then add inverse edges by introducing inverse types for each relation and interaction type. The two steps generate an augmented network in which the drugs and biomedical entities can communicate better (Fig. 1b). For a target drug pair to be predicted (for example, an emerging drug u and an existing drug v), we extract all the paths with length no longer than L between them, and combine the paths to form a path-based subgraph $\mathcal{G}^L_{u,v}$ (Fig. 1c). The value of L is a hyperparameter to be tuned (Supplementary Table 2). A flow-based GNN $g(\cdot;\theta)$ with parameters θ (Fig. 1d) is applied on $\mathcal{G}^L_{u,v}$ to trace drug features $\mathbf{h}^{(0)}_{u,u} = \mathbf{f}_u$ (like fingerprints) along the biomedical edges and integrate essential information along the path. In each iteration ℓ, the GNN flows to drug-specific entities that are ℓ steps away from drug u and $(L-\ell)$ steps away from drug v in the augmented network. An attention mechanism is applied on the edges in $\mathcal{G}^L_{u,v}$ to adjust their importance. The GNN iterates L steps to return the pairwise representation $\mathbf{h}^{(L)}_{u,v}$. Finally, $\mathbf{h}^{(L)}_{u,v}$ is fed to a linear classifier $p(\cdot)$ to predict the interaction type between u and v (Fig. 1e).

Comparison of EmerGNN to baseline methods in DDI prediction

Two public datasets, DrugBank[22] and TWOSIDES[23], were used. The original drug set was split into three parts in the ratio of 7:1:2 for training, validation and testing (Methods). The drugs in the validation and testing sets were considered emerging drugs for validation and testing, respectively. For the DrugBank dataset, there was at most one interaction type between any drug pair, and the task was to predict the exact type in a multi-type classification setting. Macro F1-score, accuracy and Cohen's kappa[24] were used as performance metrics, with F1-score the primary metric. For the TWOSIDES dataset, there may be multiple interaction types between a drug pair, and the task was to predict whether a pair of drugs would have a certain interaction type under a binary classification setting. The area under the curve of precision-recall (PR-AUC), the area under the curve of the receiver operating characteristic (ROC-AUC) and accuracy were used to evaluate the performance, with PR-AUC being the primary metric.

In the S1 setting, methods that learn drug embeddings (Emb types), particularly multidirectional semantics transmit embedding, poorly predict emerging drugs because their embeddings are not updated during training. KG-DDI performs better as it updates the drug embeddings with information in the biomedical network. Of the methods that use drug features of target drug pairs (DF-type methods), CSMDDI and STNN-DDI outperform MLP on the DrugBank dataset with their designed training schemes in a cold-start setting, but they do not perform well on TWOSIDES, with more interaction types. HIN-DDI, a method that uses graph features in the biomedical network (a GF-type method), outperforms MLP, indicating that the graph features from a biomedical network can benefit DDI prediction. Deep GNN-based methods may not perform better than DF methods on DrugBank because the GNN-based methods may not effectively capture the crucial property

https://doi.org/10.1038/s43588-023-00558-4

Table 1 | Performance of EmerGNN and other DDI prediction methods

S1 setting: DDI prediction between an emerging drug and an existing drug[a]							
Datasets		DrugBank			TWOSIDES		
Type	Methods	F1-score	Accuracy	Kappa	PR-AUC	ROC-AUC	Accuracy
DF	MLP[9]	21.1±0.8	46.6±2.1	33.4±2.5	81.5±1.5	81.2±1.9	76.0±2.1
	Similarity[4]	43.0±5.0	51.3±3.5	44.8±3.8	56.2±0.5	55.7±0.6	53.9±0.4
	CSMDDI[11]	45.5±1.8	62.6±2.8	55.0±3.2	73.2±2.6	74.2±2.9	69.9±2.2
	STNN-DDI[21]	39.7±1.8	56.7±2.6	46.5±3.4	68.9±2.0	68.3±2.6	65.3±1.8
GF	HIN-DDI*[5]	37.3±2.9	58.9±1.4	47.6±1.8	81.9±0.6	83.8±0.9	79.3±1.1
Emb	MSTE[12]	7.0±0.7	51.4±1.8	37.4±2.2	64.1±1.1	62.3±1.1	58.7±0.7
	KG-DDI*[14]	26.1±0.9	46.7±1.9	35.2±2.5	79.1±0.9	77.7±1.0	60.2±2.2
GNN	CompGCN*[28]	26.8±2.2	48.7±3.0	37.6±2.8	80.3±3.2	79.4±4.0	71.4±3.1
	Decagon*[13]	24.3±4.5	47.4±4.9	35.8±5.9	79.0±2.0	78.5±2.3	69.7±2.4
	KGNN*[16]	23.1±3.4	51.4±1.9	40.3±2.7	78.5±0.5	79.8±0.6	72.3±0.7
	SumGNN*[6]	35.0±4.3	48.8±8.2	41.1±4.7	80.3±1.1	81.4±1.0	73.0±1.4
	DeepLGF*[17]	39.7±2.3	60.7±2.4	51.0±2.6	81.4±2.1	82.2±2.6	72.8±2.8
	EmerGNN*	**62.0**±2.0	**68.6**±3.7	**62.4**±4.3	**90.6**±0.7	**91.5**±1.0	**84.6**±0.7
P value		8.9×10^{-7}	0.02	0.02	1.6×10^{-6}	6.0×10^{-8}	3.5×10^{-5}
S2 setting: DDI prediction between two emerging drugs							
Datasets		DrugBank			TWOSIDES		
Type	Methods	F1-score	Accuracy	Kappa	PR-AUC	ROC-AUC	Accuracy
DF	CSMDDI[11]	19.8±3.1	37.3±4.8	22.0±4.9	55.8±4.9	57.0±6.1	55.1±5.2
GF	HIN-DDI*[5]	8.8±1.0	27.6±2.4	13.8±2.4	64.8±2.3	58.5±1.6	59.8±1.4
Emb	KG-DDI*[14]	1.1±0.1	32.2±3.6	0.0±0.0	53.9±3.9	47.0±5.5	50.0±0.0
GNN	DeepLGF*[17]	4.8±1.9	31.9±3.7	8.2±2.3	59.4±8.7	54.7±5.9	54.0±6.2
	EmerGNN*	**25.0**±2.8	**46.3**±3.6	**31.9**±3.8	**81.4**±7.4	**79.6**±7.9	**73.0**±8.2
P value		0.02	0.01	0.01	1.4×10^{-3}	3.9×10^{-4}	7.8×10^{-3}

Four types of DDI prediction methods are compared: (1) methods that use drug features of target drug pairs (DF)[4,9,11,21]; (2) methods that use graph features in the biomedical network (GF)[5]; (3) methods that learn drug embeddings (Emb)[12,14]; (4) methods that model with GNNs (GNN)[6,13,16,17,28]. [a]All methods are run five times on fivefold datasets, with mean value and s.d. reported for the testing data. The evaluation metrics are presented as a percentage (%), with a larger value indicating better performance. Bold numbers indicate the best values and underlined numbers the second best. *P* values are computed by two-sided *t*-testing of EmerGNN over the second-best baselines. Methods leveraging a biomedical network are indicated by an asterisk.

of similarity for emerging drug prediction (Fig. 3a,b). Of these, CompGCN, Decagon and KGNN perform comparably due to their similar GNN architecture design, and SumGNN constrains message passing in the enclosing subgraph between drug pairs, making information more focused. DeepLGF is the best GNN-based baseline, as it fuses information from multiple sources and incorporates the advantages of both drug features and graph features. EmerGNN significantly outperforms all these compared methods, as indicated by the small *P* values obtained from two-sided *t*-testing of statistical significance. First, by learning paths between emerging and existing drugs, it can capture the graph features, the importance of which was verified using the GF method HIN-DDI. Second, unlike CompGCN, Decagon, KGNN and DeepLGF, the importance of edges can be weighted such that it can implicitly learn the similarity properties (Fig. 3a,b). Third, with the designed path-based subgraph and flow-based GNN architecture, EmerGNN captures more relevant information from the biomedical network, thus outperforming CompGCN and SumGNN as well (Supplementary Fig. 4).

We evaluated the top-performing models in each type in the more challenging S2 setting (Table 1), where both drugs are new, with sparser information. Although KG-DDI and DeepLGF performed well in the S1 setting, they struggled in the S2 setting because they need to learn representations of both drugs effectively. Conversely, CSMDDI and HIN-DDI performed more consistently, with CSMDDI ranking second on DrugBank and HIN-DDI ranking second on TWOSIDES. This may be due to their simple models but effective features. EmerGNN significantly outperforms all the baselines under two-sided *t*-testing of statistical significance by aggregating essential information from the biomedical network. We also provide results for the S0 setting (Supplementary Table 3), which predicts interactions between existing drugs. In the following, we thoroughly investigate why EmerGNN has superior performance for DDI prediction.

Analysis of drug interaction types in the learned subgraph

EmerGNN uses attention weights to measure the importance of edges in the subgraph for predicting the DDIs of emerging drugs. Here we analyze what is captured by the attention weights by checking the correlations between the predicted interaction types and interactions and relations in the path-based subgraphs (Fig. 3).

We first analyzed the correlations between the interaction type i_{pred} to be predicted and interaction types obtained in the selected paths. The dominant diagonal elements in the heatmap (Fig. 3a) suggest that, when predicting a target interaction i_{pred} for (u, v), paths with larger attention weights in the subgraph $\mathcal{G}^L_{u,v}$ are likely to go through another drug (for instance, u_1) that has interaction $i_1 = i_{\text{pred}}$ with the existing drug v. We suppose that drugs like u_1 may have properties similar to those of emerging drug u. To demonstrate this point, we grouped these cases of drug pairs (u, u_1) as Group 1 and other pairs (u, u_2) (with random drug u_2) as Group 2. The distributions of drug fingerprint similarities show that Group 1 contains a larger quantity of highly similar drug pairs (>0.5) than Group 2 (Fig. 3b), demonstrating

https://doi.org/10.1038/s43588-023-00558-4

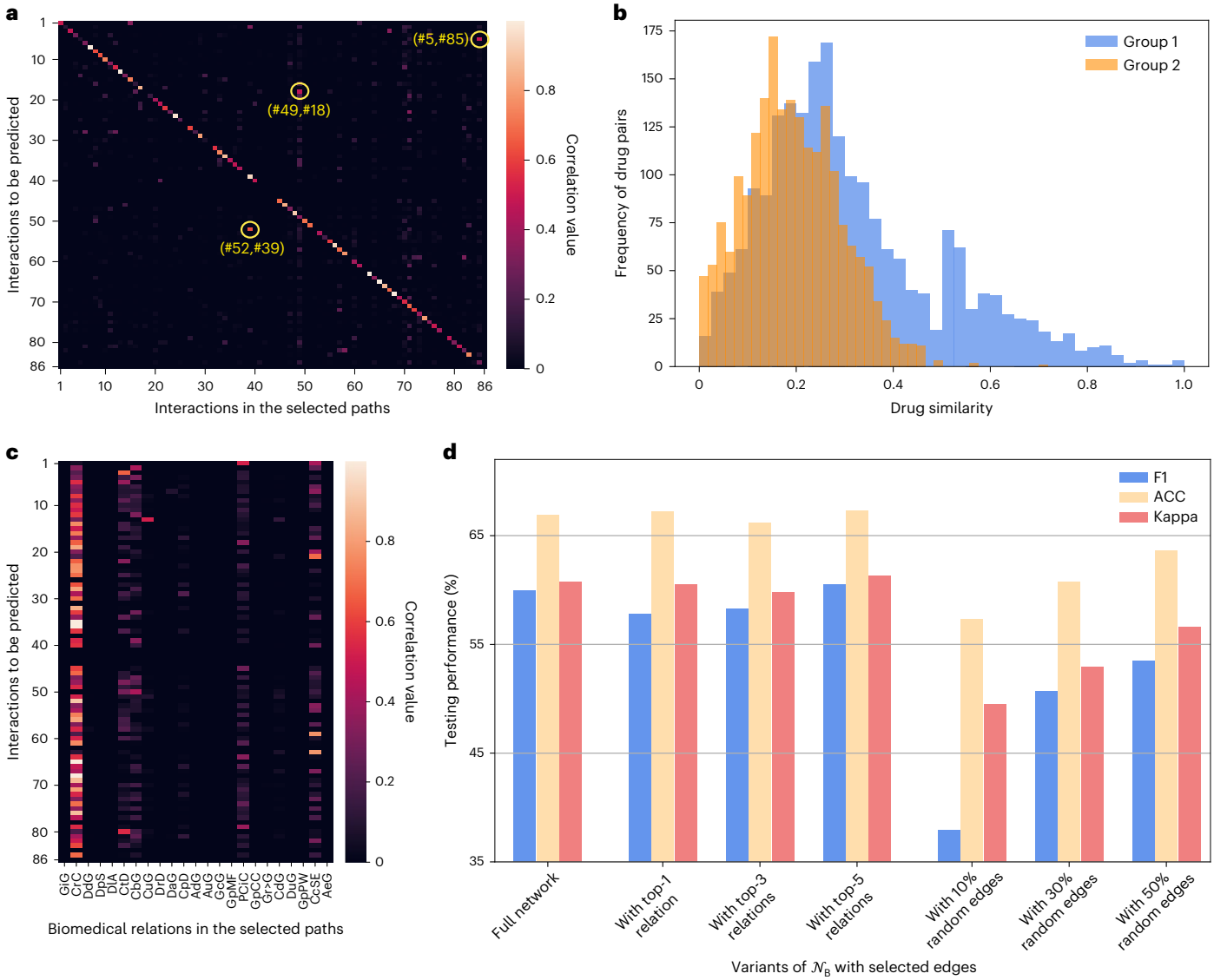

Fig. 3 | Analysis of relation types on the selected paths on the DrugBank dataset. We first extract the top five paths from u to v and from v to u, respectively, for each triplet (u, i_{pred}, v) in testing, based on the average attention weights of edges in each path, using a beam search algorithm (algorithm 2 in Supplementary Section 1). Next, we count how many times an interaction type $i \in \mathcal{R}_1$ and a relation type $r \in \mathcal{R}_B$ appear on the selected edges given i_{pred} (Supplementary Fig. 3). **a**, Heatmap of the correlation between interaction i_{pred} to be predicted and interaction types i in the selected paths. Yellow circles indicate the three interaction pairs outside the diagonal (Supplementary Table 5). **b**, Histogram distribution of fingerprint similarities in Group 1 (a drug u with another drug u_1, which connects to v with interaction type i_{pred}) and Group 2 (a drug u with a random drug u_2). **c**, Heatmap of the correlation between interaction i_{pred} to be predicted and biomedical relations r in the selected paths. **d**, Performance of the modified biomedical networks with selected relations. Leftmost is the performance of EmerGNN with a full biomedical network. The next three parts are EmerGNN with top-1 (CrC, with 0.4% edges), top-3 (CrC, CbG, CsSE, with 9.3% edges) and top-5 (CrC, CtD, CvG, PCiC, CsSE, with 9.4% edges) attended relations in the biomedical network. The rightmost three parts show the performance of EmerGNN with randomly sampled 10%, 30% and 50% edges from the biomedical network.

the crucial role of similar drugs in predicting DDIs for emerging drugs, and our method can implicitly search for these drugs. Apart from the diagonal values in the heatmap, there are strongly correlated pairs of interactions. This happens, for example, when the emerging drug u finds some connections with another drug u_3 whose intersection i_3 with the existing drug v is correlated with i_{pred}. In these cases, we find strongly correlated pairs like 'increasing constipating activity' and 'decreasing analgesic activity' (Fig. 3a and Supplementary Table 5), as verified by Liu and others[26].

We then analyzed the biomedical relation types in the selected paths by visualizing the correlations between the interaction to be predicted i_{pred} and biomedical relation types in the selected paths. A few relation types were consistently selected when predicting different interaction types (Fig. 3c). In particular, the most frequent relation type is the drug resembling relation CrC, which again verifies the importance of similar drugs for emerging drug prediction. Other frequently selected types are related to diseases (CrD), genes (CbG), pharmacologic classes (PCiC) and side effects (CsSE). To analyze their importance, we compared the performance of EmerGNN with the full biomedical network and networks with only relations having top-1, top-3 or top-5 correlation values with all the interaction types (the middle part of Fig. 3d). As a comparison, we randomly sampled 10%, 30% and 50% of edges from the biomedical performance, and the corresponding performances are shown in the right part of Fig. 3d. Keeping the top-1, top-3 and top-5 relations in the biomedical network leads to comparable performance as when using a full network.

https://doi.org/10.1038/s43588-023-00558-4

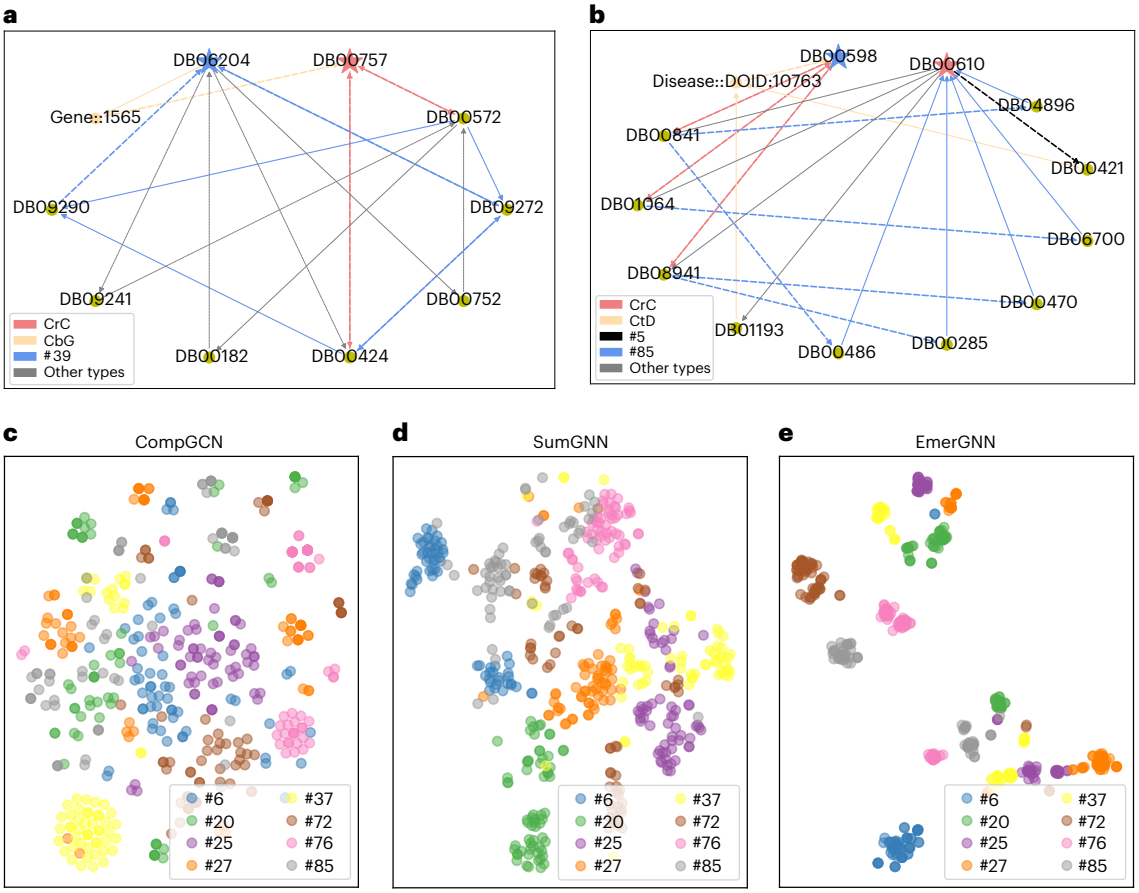

Fig. 4 | Visualization of drug pairs. a,**b**, Two cases of subgraphs containing the top ten paths according to the average of the edge attention weights on each path (explanations are provided in Supplementary Table 6). The drug pairs to be predicted are highlighted as stars, and dashed lines indicate reverse types. CrC, CbG, CtD are biomedical relations; #39, #5, #85 are interaction types; 'other types' in gray edges mean the interaction types aside from the given ones. **a**, DB06204 (tapentadol), in the blue star, is an existing drug, and DB00757 (dolasetron), in the red star, is an emerging drug. The target interaction type is '#Drug1 may decrease the analgesic activities of #Drug2' (#52). **b**, DB00598 (labetalol), in the blue star, is an emerging drug, and DB00610 (metaraminol), in the red star, is an existing drug. The target interaction type is '#Drug1 may decrease the vasoconstricting activities of #Drug2' (#5). **c**–**e**, *t*-distributed stochastic neighbor embedding visualization[37] of the representations learned for the drug pairs by CompGCN (**c**), SumGNN (**d**) and EmerGNN (**e**). As CompGCN embeds each entity separately, we concatenate embeddings of the two drugs' representations for a given drug pair. SumGNN encodes the enclosing subgraphs of (u, v) for interaction prediction, so we take the representation of the enclosing subgraph as the drug pair representation. The drug pair representation of EmerGNN is directly given by $\mathbf{h}_{u,v}^{(L)}$. Because there are too many interaction types and drug pairs in $\mathcal{N}_{\mathrm{D-test}}$, 8 interaction types and 64 drug pairs were randomly sampled for each interaction type. The legends in these figures specify the identities of the interaction type to be predicted. Each dot denotes a DDI sample (u, i, v), and the different colors in dots indicate the interaction type i that the drug pairs (u, v) have.

However, the performance substantially deteriorates when edges are randomly dropped. These experiments show that EmerGNN selects important and relevant relations in the biomedical network for DDI prediction.

Case study on drug pairs

Here we examine cases of selected paths from the subgraphs by selecting the top ten paths between u and v based on the average of the edge attention weights on each path (Fig. 4a,b). In the first case (Fig. 4a), there are interpretable paths supporting the target prediction (Supplementary Table 6). For example, there are paths connecting the two drugs through the binding protein Gene::1565 (CYP2D6), which is a P450 enzyme that plays a key role in drug metabolism[27]. Another path finds a similar drug DB00424 (hyoscyamine) for DB00757 (dolasetron) through the resemble relation (CrC), and concludes that DB06204 (tapentadol) may potentially decrease the analgesic activity of DB00757 (dolasetron) due to the correlation between constipating and analgesic activities (Fig. 3a). In the second case (Fig. 4b), we make similar observations (Supplementary Table 6). In particular, a path finds a similar drug DB00421 (spironolactone) for DB00598 (labetalol), which may decrease the vasoconstricting activity of DB00610 (metaraminol), providing a hint that labetalol may also decrease the vasoconstricting activity of metaraminol. Compared with the original subgraphs $\mathcal{G}_{u,v}^{L}$, which have tens of thousands of edges (Supplementary Fig. 1), the learned subgraphs are much smaller and more relevant to the target prediction. More examples with detailed interpretations of the paths support that EmerGNN finds important paths that indicate relevant interaction types and biomedical entities for emerging drug prediction (Supplementary Fig. 5).

Next, we visualized the drug pair representations obtained by CompGCN, SumGNN and EmerGNN (Fig. 4c–e). As shown, drug pairs with the same interaction are more densely gathered in EmerGNN than CompGCN and SumGNN. This means that the drug pair representations of EmerGNN can better separate the different interaction types. As a result, EmerGNN is able to learn better representations than other GNN methods such as CompGCN and SumGNN.

Analysis of computational complexity

Because EmerGNN learns pairwise representations for each drug pair, its computation complexity is higher than that of the other GNN-based

https://doi.org/10.1038/s43588-023-00558-4

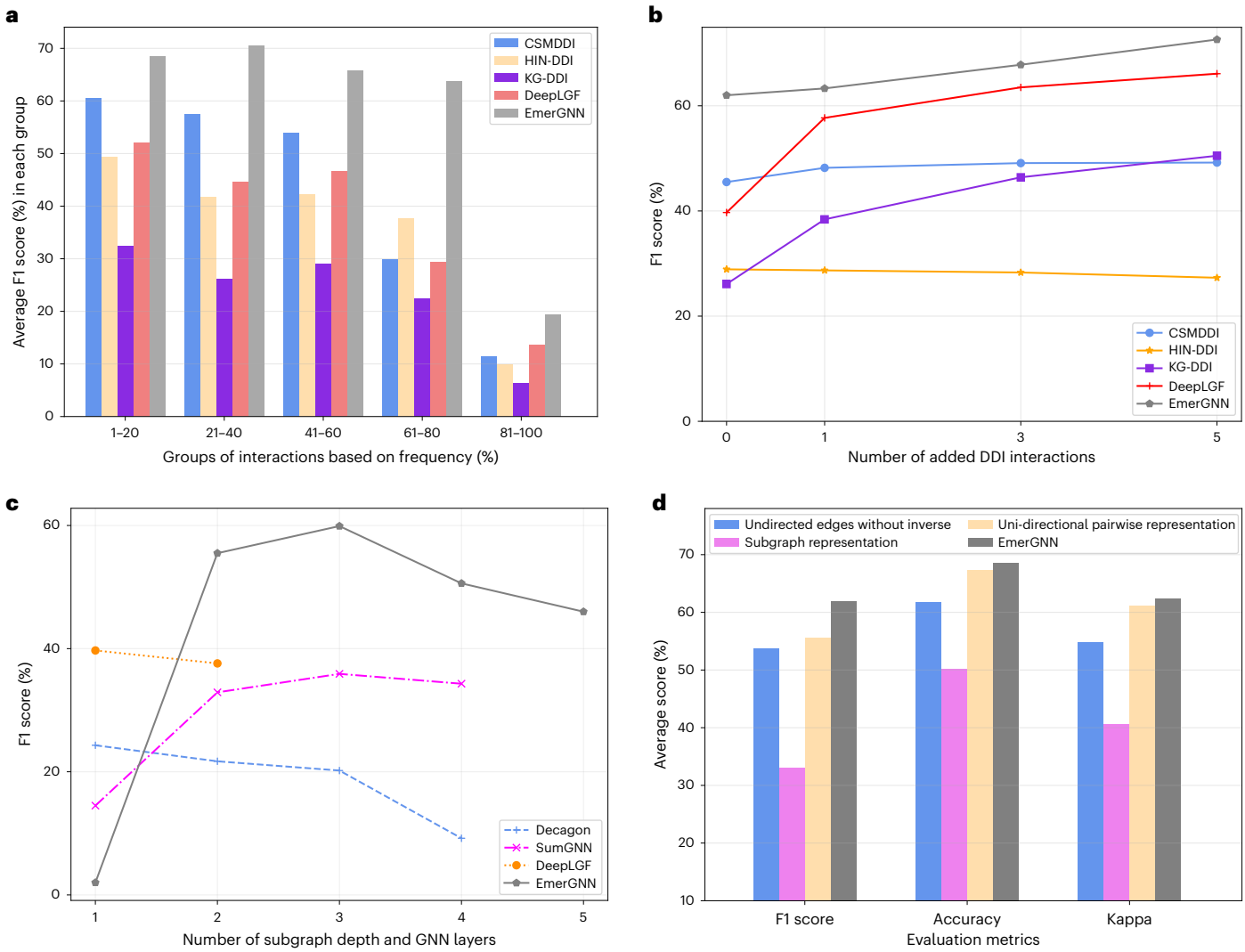

Fig. 5 | Ablation studies on the DrugBank dataset. a, Performance comparison of the interaction groups based on interaction frequency. The five groups are formed by grouping the interaction types based on their frequency in the dataset, and the average macro F1 performance is shown for each group. **b**, Performance comparison for adding interaction edges for emerging drugs into the training set, $\mathcal{N}_{\mathrm{D-train}}$. Specifically, 1/3/5 interaction edges in the testing set $\mathcal{N}_{\mathrm{D-test}}$ are randomly sampled for each emerging drug in $\mathcal{V}_{\mathrm{D-test}}$, and moved to the training set, $\mathcal{N}_{\mathrm{D-train}}$. **c**, Performance comparison for the GNN-based methods on varying depth L. Specifically, L is the number of GNN layers in Decagon and DeepLGF, the depth of the enclosing subgraph in SumGNN, and the depth of the path-based subgraph in EmerGNN. **d**, Performance comparison for the different techniques used in designing EmerGNN (Supplementary Table 7).

methods. However, EmerGNN can achieve higher accuracy than those methods in just a few hours, and a longer training time has the potential to achieve even better performance (Fig. 2a,b). Among the baseline GNN methods, Decagon is the most efficient, as it only uses information related to drug, protein and disease in the biomedical network. SumGNN and EmerGNN are slower than Decagon and DeepLGF, as they need to learn specific subgraph representations for different drug pairs. Given that the clinical development of a typical innovative drug usually takes years[25], the computation time for EmerGNN is acceptable. We also compared the graphics processing unit (GPU) memory footprint (Fig. 2c) and the number of parameters (Fig. 2d) of these GNN-based models. It is clear that EmerGNN is memory- and parameter-efficient. First, its subgraphs for DDI prediction are much smaller than the biomedical network (Supplementary Fig. 1). Second, EmerGNN mainly relies on the biomedical concepts instead of the drugs' embeddings to carry out predictions, resulting in a small number of parameters. In comparison, DeepLGF requires a large number of model parameters to learn embeddings from the biomedical network.

Ablation studies

We compared the performances of the top-performing models according to the frequency of interaction types to analyze the different models' abilities (Fig. 5a). EmerGNN outperformed the baselines for all frequencies. For the high-frequency relations (~1–20%), all the methods, except for KG-DDI, gave a good performance. For extremely low-frequency relations (~81–100%), all the methods worked poorly. The performances of all the methods deteriorated in general for relations with a lower frequency. However, the relative performance gain of EmerGNN tends to be larger, especially in the range ~61–80%. These results indicate EmerGNN's strengths in generalization and the ability to extract essential information from the biomedical network for predicting rare drugs and interaction types.

The main experiments (Table 1) study the scenario of emerging drugs without any interaction with existing drugs. In practice, we may have a few known interactions between the emerging and existing drugs, often obtained from limited clinical trials. Hence, we analyze how different models perform if adding a few interactions for each emerging drug (Fig. 5b). We can see that the performance of shallow

models such as CSMDDI and HIN-DNN does not change much, because the features they use are unchanged. However, methods learning drug embeddings, such as KG-DDI and DeepLGF, enjoy more substantial improvement when additional knowledge is provided. In comparison, EmerGNN has increased performance with more interactions added and still performs best compared with all the other methods.

The value of L determines the maximum number of hops of neighboring entities that the GNN-based models can visit. We studied the impact of changing L for these methods (Fig. 5c). The performance of Decagon and DeepLGF gets worse as L becomes larger. As Decagon and DeepLGF work on the full biomedical network, too much irrelevant information will be involved in the representation learning, leading to worse performance. DeepLGF runs out-of-memory when $L \geq 3$. SumGNN and EmerGNN perform worst with $L = 1$, as it is hard for the information to be passed from the emerging drug to the existing drug. SumGNN can leverage the drug features for prediction, and thus outperforms Decagon. In comparison, EmerGNN benefits to a large extent from the relevant information on the biomedical network when L increases from 1 to 3. However, the performance decreases when $L > 3$. Intuitively, the path-based subgraph contains too much irrelevant information when the length is longer, increasing the learning difficulty. Hence, a moderate path length, $L = 3$, is optimal for EmerGNN, when considering the effectiveness and computation efficiency.

We conducted experiments to analyze the main techniques used in designing EmerGNN (Fig. 5d). First, we evaluated the performance when using undirected edges without introducing inverse edges (denoted 'undirected edges without inverse'). It is clear that using undirected edges has a negative effect as the directional information on the biomedical network is lost. We then designed a variant that learns a subgraph representation as SumGNN upon $\mathcal{G}^L_{u,v}$ (denoted 'subgraph representation'), and another variant that only learns on the uni-directional computing (Methods) from direction u to v without considering the direction v to u (denoted as 'uni-directional pairwise representation'). Comparing subgraph representation with uni-directional pairwise representation, we observe that the flow-based GNN architecture is more effective than the GNN used in SumGNN. Even though uni-directional pairwise representation can achieve better performance compared with all the baselines in the S1 setting (Table 1), learning bi-directional representations can help to further improve the prediction ability by balancing the bi-directional communications between drugs.

Discussion

Predicting DDIs for emerging drugs is a crucial issue in biomedical computational science as it offers possibilities for treating and alleviating diseases. Recent advances have been made in DDI prediction accuracy through the use of deep neural networks[5,13,14,17,19,20], but these methods require a large amount of known DDI information, which is often scarce for emerging drugs. Additionally, some approaches designed for DDI prediction only leverage shallow features, limiting their expressiveness in this task.

One limitation of EmerGNN is that the emerging drug to be predicted should be included in the biomedical network. Building connections between emerging drugs and existing drugs through molecular formulas or properties may help address this issue. Although we demonstrate the effectiveness of EmerGNN for DDI prediction in this Article, EmerGNN is a general approach that can be applied to other biomedical applications, such as predicting protein–protein interactions, drug–target interactions and disease–gene interactions. We anticipate that the paths selected according to the attention values on edges by EmerGNN can enhance the accuracy and interpretability of these predictions. We hope that our open-source EmerGNN can serve as a strong deep learning tool to advance biomedicine and healthcare, by enabling practitioners to exploit the rich knowledge in existing large biomedical networks for low-data scenarios.

Methods

To predict the interactions between emerging drugs and existing drugs, it is important to leverage relevant information in the biomedical network. Our framework has four main elements: (1) constructing an augmented network by integrating the DDI network with the biomedical network and adding inverse edges; (2) extracting all paths with length no longer than L from u to v to construct a path-based subgraph $\mathcal{G}^L_{u,v}$; (3) encoding the pairwise subgraph representation $\mathbf{h}^{(L)}_{u,v}$ by a flow-based GNN with an attention mechanism such that the information can flow from u over the important entities and edges in $\mathcal{G}^L_{u,v}$ to v; (4) predicting the interaction type based on the bi-directional pairwise subgraph representations. The overall framework is shown in Fig. 1.

Augmented network

Given the DDI network $\mathcal{N}_\mathrm{D} = \{(u, i, v) : u, v \in \mathcal{V}_\mathrm{D}, i \in \mathcal{R}_\mathrm{I}\}$ and the biomedical network $\mathcal{N}_\mathrm{B} = \{(h, r, t) : h, t \in \mathcal{V}_\mathrm{B}, r \in \mathcal{R}_\mathrm{B}\}$ ($\mathcal{N}_\mathrm{D}$ is specified as $\mathcal{N}_{\mathrm{D-train}}$ / $\mathcal{N}_{\mathrm{D-valid}}$/ $\mathcal{N}_{\mathrm{D-test}}$ in the training/validation/testing stages, respectively), we integrate the two networks into

$$\mathcal{N}' = \mathcal{N}_\mathrm{D} \cup \mathcal{N}_\mathrm{B} = \{(e, r, e') : e, e' = \mathcal{V}', r \in \mathcal{R}'\}$$

with $\mathcal{V}' = \mathcal{V}_\mathrm{D} \cup \mathcal{V}_\mathrm{B}$ and $\mathcal{R}' = \mathcal{R}_\mathrm{I} \cup \mathcal{R}_\mathrm{B}$. The integrated network $\mathcal{N}'$ connects the existing and emerging drugs by concepts in the biomedical network. Because the relation types are directed, we follow common practices in knowledge graph learning[6,28] to add inverse types. Specifically, we add r_inv for each $r \in \mathcal{R}'$ and create a set of inverse types $\mathcal{R}'_\mathrm{inv}$, which subsequently leads to an inverse network

$$\mathcal{N}'_\mathrm{inv} = \{(e', r_\mathrm{inv}, e) : (e, r, e') \in \mathcal{N}'\}$$

Note that the inverse relations will not influence the information in the original biomedical network as we can transform any inverse edge (e', r_inv, e) back to the original edge (e, r, e'). Semantically, the inverse relations can be regarded as a kind of active voice versus passive voice in linguistics, for instance includes_inv can be regarded as 'being included' and causes_inv can be regarded as 'being caused'. By adding the inverse edges, the paths can be smoothly organized in single directions. For example, a path $a \xrightarrow{r1} b \xleftarrow{r2} c$ can be transformed to $a \xrightarrow{r1} b \xrightarrow{r2\_\mathrm{inv}} c$, which is more computational friendly.

After the above two steps, we obtain the augmented network

$$\mathcal{N} = \mathcal{N}' \cup \mathcal{N}'_\mathrm{inv} = \{(e, r, e') : e, e' \in \mathcal{V}, r \in \mathcal{R}\}$$

with entity set $\mathcal{V} = \mathcal{V}' = \mathcal{V}_\mathrm{D} \cup \mathcal{V}_\mathrm{B}$ and relation set $\mathcal{R} = \mathcal{R}' \cup \mathcal{R}'_\mathrm{inv}$.

Path-based subgraph formulation

Inspired by the path-based methods in knowledge graph learning[29,30], we were motivated to extract the paths connecting existing and emerging drugs, and predict the interaction type based on the paths.

Given a drug pair (u, v) to be predicted, we extract the set $\mathcal{P}^L_{u,v}$ of all the paths with length up to L. Each path in $\mathcal{P}^L_{u,v}$ has the form

$$e_0 \xrightarrow{r_1} e_1 \xrightarrow{r_2} \cdots \xrightarrow{r_L} e_L$$

with $e_0 = u, e_L = v$ and $(e_{i-1}, r_i, e_i) \in \mathcal{N}, i = 1, \ldots, L$. The intermediate entities $e_1, \ldots, e_{L-1} \in \mathcal{V}$ can be drugs, genes, diseases, side effects, symptoms, pharmacologic class and so on, and $r_1, \ldots, r_L \in \mathcal{R}$ are the interactions or relations between the biomedical entities. To preserve the local structures, we merge the paths in $\mathcal{P}^L_{u,v}$ to a subgraph $\mathcal{G}^L_{u,v}$ such that the same entities are merged to a single node. The detailed steps of path extraction and subgraph generation are provided in Supplementary Section 1.

Unlike the subgraph structures used for link prediction on general graphs[6,31,32], the edges in $\mathcal{G}^L_{u,v}$ are pointed away from u and towards v.

Our objective is to learn a GNN $g(\cdot)$ with parameters θ that predicts the DDI between u and v based on the path-based subgraph $\mathcal{G}^L_{u,v}$, that is

$$\mathrm{DDI}(u, v) = g(\mathcal{G}^L_{u,v}; \theta) \quad (1)$$

The link prediction problem on the DDI network is then transformed as a whole graph learning problem.

Flow-based GNN architecture

Given $\mathcal{G}^L_{u,v}$, we would like to integrate essential information in it to predict the target interaction type. Note that the edges in $\mathcal{G}^L_{u,v}$ are from the paths $\mathcal{P}^L_{u,v}$ connecting from u to v. We aim to design a special GNN architecture such that the information can flow from drug u to v, via integrating entities and relations in $\mathcal{G}^L_{u,v}$.

Denote $\mathcal{V}^\ell_{u,v}$, $\ell = 0, \ldots, L$, as the set of entities that can be visited in the ℓth flow step from u (like the four ellipses in $g(\mathcal{G}^L_{u,v}; \theta)$ in Fig. 1). In particular, we have $\mathcal{V}^0_{u,v} = \{u\}$ as the starting point and $\mathcal{V}^L_{u,v} = \{v\}$ as the end point. In the ℓth iteration, the visible entities in $\mathcal{V}^\ell_{u,v}$ contain entities that are ℓ steps away from drug u and $(L-\ell)$-steps away from drug v in the augmented network $\mathcal{N}$. We use the fingerprint features[9] of drug u as the input representation of u, namely $\mathbf{h}^{(0)}_{u,u} = \mathbf{f}_u$. We then conduct message flow for L steps with the function

$$\mathbf{h}^{(\ell)}_{u,e} = \delta\left(W^{(\ell)} \sum_{e' \in \mathcal{V}^{\ell-1}_{u,v}} \left(\mathbf{h}^{(\ell-1)}_{u,e} + \phi(\mathbf{h}^{(\ell-1)}_{u,e'}, \mathbf{h}^{(\ell)}_r)\right)\right) \quad (2)$$

for entities $e \in \mathcal{V}^\ell_{u,v}$, where $W^{(\ell)} \in \mathbb{R}^{d\times d}$ is a learnable weighting matrix for step ℓ, $\mathbf{h}^{(\ell-1)}_{u,e'}$ is the pairwise representation of entity $e' \in \mathcal{V}^{\ell-1}_{u,v}$, r is the relation type between e' and e, $\mathbf{h}^{(\ell)}_r \in \mathbb{R}^d$ is the learnable representation with dimension d of r in the ℓth step, $\phi(\cdot,\cdot) : (\mathbb{R}^d, \mathbb{R}^d) \to \mathbb{R}^d$ is the function combining the two vectors, and $\delta(\cdot)$ is the activation function ReLU[33].

Because the biomedical network is not specially designed for the DDI prediction task, we need to control the importance of different edges in $\mathcal{G}^L_{u,v}$. We use a drug-dependent attention weight for function $\phi(\cdot,\cdot)$. Specifically, we design the message function for each edge (e', r, e) during the ℓth propagation step as

$$\phi(\mathbf{h}^{(\ell-1)}_{u,e'}, \mathbf{h}^{(\ell)}_r) = \alpha^{(\ell)}_r \cdot \left(\mathbf{h}^{(\ell-1)}_{u,e'} \odot \mathbf{h}^{(\ell)}_r\right) \quad (3)$$

where $\odot$ is an element-wise dot product of vectors and $\alpha^{(\ell)}_r$ is the attention weight controlling the importance of messages. We design the attention weight depending on the edges' relation type as

$$\alpha^{(\ell)}_r = \sigma\left((\mathbf{w}^{(\ell)}_r)^\top [\mathbf{f}_u; \mathbf{f}_v]\right)$$

where the relation weight $\mathbf{w}^{(\ell)}_r \in \mathbb{R}^{2d}$ is multiplied by the fingerprints $[\mathbf{f}_u; \mathbf{f}_v] \in \mathbb{R}^{2d}$ of drugs to be predicted, and $\sigma(\cdot)$ is a sigmoid function returning a value in (0, 1).

After iterating for L steps, we can obtain the representation $\mathbf{h}^{(L)}_{u,v}$ that encodes the important paths up to length L between drugs u and v.

Objective and training

In practice, the interaction types can be symmetric, for example #Drug1 and #Drug2 may have the side effect of headache if used together, or asymmetric, for example #Drug1 may decrease the analgesic activities of #Drug2. Furthermore, the emerging drug can appear in either the source (drug u) or target (drug v). We extract the reverse subgraph $\mathcal{G}^L_{v,u}$ and encode it with the same parameters in equation (2) to obtain the reverse pairwise representation $\mathbf{h}^{(L)}_{v,u}$. The bi-directional representations are then concatenated to predict the interaction type with

$$\mathbf{l}(u, v) = W_{\mathrm{rel}}\left[\mathbf{h}^{(L)}_{u,v}; \mathbf{h}^{(L)}_{v,u}\right] \quad (4)$$

Here, the transformation matrix $W_{\mathrm{rel}} \in \mathbb{R}^{|\mathcal{R}_1|\times 2d}$ is used to map the pairwise representations into prediction logits $\mathbf{l}(u, v)$ of the $|\mathcal{R}_1|$ interaction types. The ith logit $l_i(u, v)$ indicates the plausibility of interaction type i being predicted. The full algorithm and implementation details of equation (4) are provided in Supplementary Section 1.

Because we have two kinds of task that are multi-class (on the DrugBank dataset) and multi-label (on the TWOSIDES dataset) interaction predictions, the training objectives are different.

For DrugBank, there exists at most one interaction type between two drugs. Given two drugs u and v, once we obtain the prediction logits $\mathbf{l}(u, v)$ of different interaction types, we use a softmax function to compute the probability of each interaction type, namely

$$I_i(u, v) = \frac{\exp(l_i(u, v))}{\sum_{j\in\mathcal{R}_1} \exp(l_j(u, v))}$$

Denote $\mathbf{y}(u, v) \in \mathbb{R}^{|\mathcal{R}_1|}$ as the ground-truth indicator of the target interaction type, where $y_i(u, v) = 1$ if $(u, i, v) \in \mathcal{N}_{\mathrm{D}}$, otherwise zero. We minimize the following cross-entropy loss to train the model parameters:

$$\mathcal{L}_{\mathrm{DB}} = -\sum_{(u,i,v)\in\mathcal{N}_{\mathrm{D-train}}} y_i(u, v) \log I_i(u, v). \quad (5)$$

For TWOSIDES, there may be multiple interactions between two drugs. The objective is to predict whether there is an interaction p between two drugs. Given two drugs u, v and the prediction logits $\mathbf{l}(u, v)$, we use the sigmoid function

$$I_i(u, v) = \frac{1}{1 + \exp(-l_i(u, v))}$$

to compute the probability of interaction type i. Unlike the multi-class task in DrugBank, we use the binary cross-entropy loss

$$\mathcal{L}_{\mathrm{TS}} = -\sum_{(u,i,v)\in\mathcal{N}_{\mathrm{D-train}}} \left(\log(I_i(u, v)) + \sum_{(u',v')\in\mathcal{N}_i} \log(1 - I_i(u', v'))\right) \quad (6)$$

where $\mathcal{N}_i$ is the set of drug pairs that do not have the interaction type i.

We use the stochastic gradient optimizer Adam[34] to optimize the model parameters

$$\theta = \left\{W_{\mathrm{rel}}, \{W^{(\ell)}, \mathbf{h}^{(\ell)}_r, \mathbf{w}^{(\ell)}_r\}_{\ell=1,\ldots,L,r\in\mathcal{R}}\right\}$$

by minimizing the loss function in equation (5) for the DrugBank dataset or equation (6) for the TWOSIDES dataset.

DDI network

Following refs. 6,13, we used two benchmark datasets, DrugBank[22] and TWOSIDES[23], as the interaction network $\mathcal{N}_{\mathrm{D}}$ (Supplementary Table 1). When predicting DDIs for emerging drugs, namely the S1 and S2 settings, we randomly split $\mathcal{V}_{\mathrm{D}}$ into three disjoint sets with $\mathcal{V}_{\mathrm{D}} = \mathcal{V}_{\mathrm{D-train}} \cup \mathcal{V}_{\mathrm{D-valid}} \cup \mathcal{V}_{\mathrm{D-test}}$ and $\mathcal{V}_{\mathrm{D-train}} \cap \mathcal{V}_{\mathrm{D-valid}} \cap \mathcal{V}_{\mathrm{D-test}} = \varnothing$, where $\mathcal{V}_{\mathrm{D-train}}$ is the set of existing drugs used for training, $\mathcal{V}_{\mathrm{D-valid}}$ is the set of emerging drugs for validation, and $\mathcal{V}_{\mathrm{D-test}}$ is the set of emerging drugs for testing. The interaction network for training is defined as $\mathcal{N}_{\mathrm{D-train}} = \{(u, i, v) \in \mathcal{N}_{\mathrm{D}} : u, v \in \mathcal{V}_{\mathrm{D-train}}\}$.

In the S1 setting, we set

$\mathcal{N}_{\mathrm{D-valid}} = \{(u, i, v) \in \mathcal{N}_{\mathrm{D}} : u \in \mathcal{V}_{\mathrm{D-train}}, v \in \mathcal{V}_{\mathrm{D-valid}}\} \cup \{(u, i, v) \in \mathcal{N}_{\mathrm{D}} : u \in \mathcal{V}_{\mathrm{D-valid}}, v \in \mathcal{V}_{\mathrm{D-train}}\}$

as validation samples, and

$\mathcal{N}_{\mathrm{D-test}} = \{(u, i, v) \in \mathcal{N}_{\mathrm{D}} : u \in (\mathcal{V}_{\mathrm{D-train}} \cup \mathcal{V}_{\mathrm{D-valid}}), v \in \mathcal{V}_{\mathrm{D-test}}\} \cup \{(u, i, v) \in \mathcal{N}_{\mathrm{D}} : u \in \mathcal{V}_{\mathrm{D-test}}, v \in (\mathcal{V}_{\mathrm{D-train}} \cup \mathcal{V}_{\mathrm{D-valid}})\}$ as testing samples.

In the S2 setting, we set

$\mathcal{N}_{\mathrm{D-valid}} = \{(u, i, v) \in \mathcal{N}_{\mathrm{D}} : u, v \in \mathcal{V}_{\mathrm{D-valid}}\}$ as validation samples, and $\mathcal{N}_{\mathrm{D-test}} = \{(u, i, v) \in \mathcal{N}_{\mathrm{D}} : u, v \in \mathcal{V}_{\mathrm{D-test}}\}$ as testing samples.

We followed ref. 6 to randomly sample one negative sample for each $(u, i, v) \in \mathcal{N}_{\mathrm{D-valid}} \cup \mathcal{N}_{\mathrm{D-test}}$ to form the negative set $\mathcal{N}_i$ for the TWOSIDES dataset in the evaluation phase. Specifically, if u is an emerging drug, we randomly sample an existing drug $v' \in \mathcal{V}_{\mathrm{D-train}}$ and make sure that the new interaction does not exist, namely $(u, i, v') \notin \mathcal{N}_{\mathrm{D}}$. If v is an emerging drug, we randomly sample an existing drug $u' \in \mathcal{V}_{\mathrm{D-train}}$ and make sure that the new interaction does not exist, namely $(u', i, v) \notin \mathcal{N}_{\mathrm{D}}$.

Biomedical network

Here, as for the DDI network, we use different variants of the biomedical network $\mathcal{N}_{\mathrm{B}}$ for training, validation and testing. The well-constructed biomedical network HetioNet[18] was used here. We denote $\mathcal{V}_{\mathrm{B}}$, $\mathcal{R}_{\mathrm{B}}$, $\mathcal{N}_{\mathrm{B}}$ as the set of entities, relations and edges, respectively, in the full biomedical network. When predicting interactions between existing drugs in the S0 setting, all the edges in $\mathcal{N}_{\mathrm{B}}$ are used for training, validation and testing. When predicting interactions between emerging drugs and existing drugs (S1 and S2 settings), we use different parts of the biomedical networks.

To guarantee that the emerging drugs are connected with some existing drugs through the biomedical entities, we constrain the split of drugs to satisfy the conditions $\mathcal{V}_{\mathrm{D-valid}} \subset \mathcal{V}_{\mathrm{B}}$ and $\mathcal{V}_{\mathrm{D-test}} \subset \mathcal{V}_{\mathrm{B}}$. Meanwhile, we also guarantee that the emerging drugs will not be seen in the biomedical network during training. To achieve this goal, the edges for training are in the set $\mathcal{N}_{\mathrm{B-train}} = \{(h, r, t) \in \mathcal{N}_{\mathrm{B}} : h, t \notin (\mathcal{V}_{\mathrm{D-valid}} \cup \mathcal{V}_{\mathrm{D-test}})\}$, the edges for validation are in the set $\mathcal{N}_{\mathrm{B-valid}} = \{(h, r, t) \in \mathcal{N}_{\mathrm{B}} : h, t \notin \mathcal{V}_{\mathrm{D-test}}\}$, and the testing network is the original network, namely $\mathcal{N}_{\mathrm{B-test}} = \mathcal{N}_{\mathrm{B}}$.

In addition, we plotted the size distribution (measured by the number of edges in $\mathcal{G}^L_{u,v}$) as histograms (Supplementary Fig. 1). We observe that both datasets follow long-tailed distributions. Many subgraphs have tens of thousands of edges on DrugBank, but hundreds of thousands of edges on TWOSIDES, because the DDI network is denser. However, compared with the augmented networks, which have sizes of 3,657,114 for DrugBank and 3,567,059 for TWOSIDES, the sizes of the subgraphs are quite small.

Evaluation metrics

As suggested by ref. 6, there is, at most, one interaction between a pair of drugs in the DrugBank dataset[22]. Hence, we evaluated the performance in a multi-class setting, which estimates whether the model can correctly predict the interaction type for a pair of drugs. We considered the following metrics:

- F1(macro) $= \frac{1}{\|\mathcal{T}_{\mathrm{D}}\|} \sum_{i \in \mathcal{T}_{\mathrm{D}}} \frac{2P_i \cdot R_i}{P_i + R_i}$, where P_i and R_i are the precision and recall for interaction type i, respectively. Macro F1 aggregates the fractions over different interaction types.
- Accuracy, the percentage of correctly predicted interaction type compared with the ground-truth interaction type.
- Cohen's kappa[24], $\kappa = \frac{A_p - A_e}{1 - A_e}$, where A_p is the observed agreement (accuracy) and A_e is the probability of randomly seeing each class.

In the TWOSIDES dataset[23], there may be multiple interactions between a pair of drugs, such as anemia, nausea and pain. Hence, we modeled and evaluated the performance in a multi-label setting, where each type of side effect is modeled as a binary classification problem. Following refs. 13,23, we sample one negative drug pair for each $(u, i, v) \in \mathcal{N}_{\mathrm{D-test}}$ and evaluate the binary classification performance with the following metrics:

- ROC-AUC, the area under the curve of receiver operating characteristics, measured by $\sum_{k=1}^{n} \mathrm{TP}_k \Delta \mathrm{FP}_k$, where $(\mathrm{TP}_k, \mathrm{FP}_k)$ is the true-positive and false-positive of the kth operating point.
- PR-AUC, the area under curve of precision-recall, measured according to $\sum_{k=1}^{n} P_k \Delta R_k$, where (P_k, R_k) is the precision and recall of the kth operating point.
- Accuracy, the average precision of drug pairs for each side effect.

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

The resplit dataset[35] in DrugBank, TWOSIDES and HetioNet for the S1 and S2 settings is publicly available at https://doi.org/10.5281/zenodo.10016715. Source data are provided with this paper.

Code availability

The code for EmerGNN[36] is available at https://github.com/LARS-research/EmerGNN.

References

1. Su, X., Wang, H., Zhao, N., Wang, T. & Cui, Y. Trends in innovative drug development in China. *Nat. Rev. Drug Discov.* **21**, 709–710 (2022).
2. Ledford, H. Hundreds of COVID trials could provide a deluge of new drugs. *Nature* **603**, 25–27 (2022).
3. Percha, B. & Altman, R. B. Informatics confronts drug-drug interactions. *Trends Pharmacol. Sci.* **34**, 178–184 (2013).
4. Vilar, S. et al. Similarity-based modeling in large-scale prediction of drug-drug interactions. *Nat. Protoc.* **9**, 2147–2163 (2014).
5. Tanvir, F., Islam, M. I. K. & Akbas, E. Predicting drug-drug interactions using meta-path based similarities. In *Proc. IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology* (eds Hallinan, J. et al.) 1–8 (IEEE, 2021).
6. Yu, Y. et al. SumGNN: multi-typed drug interaction prediction via efficient knowledge graph summarization. *Bioinformatics* **37**, 2988–2995 (2021).
7. Letinier, L. et al. Risk of drug–drug interactions in out-hospital drug dispensings in France: results from the drug–drug interaction prevalence study. *Front. Pharmacol.* **10**, 265 (2019).
8. Jiang, H. et al. Adverse drug reactions and correlations with drug–drug interactions: a retrospective study of reports from 2011 to 2020. *Front. Pharmacol.* **13**, 923939 (2022).
9. Rogers, D. & Hahn, M. Extended-connectivity fingerprints. *J. Chem. Inf. Model.* **50**, 742–754 (2010).
10. Dewulf, P., Stock, M. & De Baets, B. Cold-start problems in data-driven prediction of drug-drug interaction effects. *Pharmaceuticals* **14**, 429 (2021).
11. Liu, Z., Wang, X.-N., Yu, H., Shi, J.-Y. & Dong, W.-M. Predict multi-type drug-drug interactions in cold start scenario. *BMC Bioinformatics* **23**, 75 (2022).
12. Yao, J., Sun, W., Jian, Z., Wu, Q. & Wang, X. Effective knowledge graph embeddings based on multidirectional semantics relations for polypharmacy side effects prediction. *Bioinformatics* **38**, 2315–2322 (2022).
13. Zitnik, M., Agrawal, M. & Leskovec, J. Modeling polypharmacy side effects with graph convolutional networks. *Bioinformatics* **34**, i457–i466 (2018).
14. Karim, M. R. et al. Drug–drug interaction prediction based on knowledge graph embeddings and convolutional-LSTM network. In *Proc. 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics* (eds Shi, X. & Buck, M.) 113–123 (Association for Computing Machinery, 2019).
15. Huang, K., Xiao, C., Glass, L. M., Zitnik, M. & Sun, J. SkipGNN: predicting molecular interactions with skip-graph networks. *Sci. Rep.* **10**, 21092 (2020).
16. Lin, X., Quan, Z., Wang, Z.-J., Ma, T. & Zeng, X. KGNN: knowledge graph neural network for drug-drug interaction prediction. In *Proc. Twenty-Ninth International Joint Conference on Artificial Intelligence* (ed. Bessiere, C.) 2739–2745 (IJCAI, 2020).

17. Ren, Z.-H. et al. A biomedical knowledge graph-based method for drug-drug interactions prediction through combining local and global features with deep neural networks. *Brief. Bioinformatics* **23**, bbac363 (2022).
18. Himmelstein, D. S. et al. Systematic integration of biomedical knowledge prioritizes drugs for repurposing. *eLife* **6**, e26726 (2017).
19. Kipf, T. N. & Welling, M. Semi-supervised classification with graph convolutional networks. In *Proc. 5th International Conference on Learning Representations* https://openreview.net/forum?id=SJU4ayYgl (OpenReview.net, 2017).
20. Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O. & Dahl, G. E. Neural message passing for quantum chemistry. In *International Conference on Machine Learning* (eds Precup, D. & Teh, Y. W.) 1263–1272 (Association for Computing Machinery, 2017).
21. Yu, H., Zhao, S. Y. & Shi, J. Y. STNN-DDI: a substructure-aware tensor neural network to predict drug-drug interactions. *Brief. Bioinformatics* **23**, bbac209 (2022).
22. Wishart, D. S. et al. DrugBank 5.0: a major update to the DrugBank database for 2018. *Nucleic Acids Res.* **46**, D1074–D1082 (2018).
23. Tatonetti, N. P., Ye, P. P., Daneshjou, R. & Altman, R. B. Data-driven prediction of drug effects and interactions. *Sci. Transl. Med.* **4**, 125ra31–125ra31 (2012).
24. Cohen, J. A coefficient of agreement for nominal scales. *Educ. Psychol. Meas.* **20**, 37–46 (1960).
25. Brown, D. G., Wobst, H. J., Kapoor, A., Kenna, L. A. & Southall, N. Clinical development times for innovative drugs. *Nat. Rev. Drug Discov.* **21**, 793–794 (2021).
26. Liu, M. & Wittbrodt, E. Low-dose oral naloxone reverses opioid-induced constipation and analgesia. *J. Pain Symptom Manag.* **23**, 48–53 (2002).
27. Estabrook, R. W. A passion for P450s (remembrances of the early history of research on cytochrome P450). *Drug Metab. Dispos.* **31**, 1461–1473 (2003).
28. Vashishth, S., Sanyal, S., Nitin, V. & Talukdar, P. Composition-based multi-relational graph convolutional networks. In *Proc. 8th International Conference on Learning Representations* https://openreview.net/pdf?id=BylA_C4tPr (OpenReview.net, 2020).
29. Lao, N., Mitchell, T. & Cohen, W. Random walk inference and learning in a large scale knowledge base. In *Proc. 2011 Conference on Empirical Methods in Natural Language Processing* (eds Merlo, P. & Barzilay, R.) 529–539 (Association for Computing Machinery, 2011).
30. Xiong, W., Hoang, T. & Wang, W. Y. DeepPath: a reinforcement learning method for knowledge graph reasoning. In *Proc. 2017 Conference on Empirical Methods in Natural Language Processing* (eds Specia, L. et al.) 564–573 (Association for Computational Linguistics, 2017).
31. Zhang, M. & Chen, Y. Link prediction based on graph neural networks. In *Proc. 32nd International Conference on Neural Information Processing Systems* (eds Bengio, S. & Wallach, H. M.) 5171–5181 (Association for Computing Machinery, 2018).
32. Teru, K., Denis, E. & Hamilton, W. Inductive relation prediction by subgraph reasoning. In *International Conference on Machine Learning* (eds Daumé III, H. & Singh, A.) 9448–9457 (Association for Computing Machinery, 2020).
33. Nair, V. & Hinton, G. E. Rectified linear units improve restricted Boltzmann machines. In *Proc. 27th International Conference on Machine Learning* (eds Joachims, T. & Furnkranz, J.) 807–814 (Association for Computing Machinery, 2010).
34. Kingma, D. P & Ba, J. Adam: a method for stochastic optimization. In *Proc. 3rd International Conference on Learning Representations* (eds Bengio, Y. & LeCun, Y.) https://arxiv.org/pdf/1412.6980.pdf (2014).
35. Zhang, Y., Yue, L. & Yao, Q. EmerGNN_DDI_data. *Zenodo* https://doi.org/10.5281/zenodo.10016715 (2023).
36. Zhang, Y., Yue, L. & Yao, Q. LARS-research/EmerGNN: v1.0.0k. *Zenodo* https://doi.org/10.5281/zenodo.10017431 (2023).
37. Van der Maaten, L. & Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **9**, 2579–2605 (2008).

Acknowledgements

This project was supported by the National Natural Science Foundation of China (no. 92270106) and the CCF-Tencent Open Research Fund.

Author contributions

Y. Zhang contributed to idea development, algorithm implementation, experimental design, results analysis and writing of the paper. Q.Y. contributed to idea development, experimental design, results analysis and writing of the paper. L.Y. contributed to algorithm implementation and results analysis. Y. Zheng contributed to results analysis and writing of the paper. All authors read, edited and approved the paper.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s43588-023-00558-4.

Correspondence and requests for materials should be addressed to Quanming Yao.

Peer review information *Nature Computational Science* thanks Nguyen Quoc Khanh Le, Jian-Yu Shi and the other, anonymous, reviewer(s) for their contribution to the peer review of this work. Peer reviewer reports are available. Primary Handling Editor: Kaitlin McCardle, in collaboration with the *Nature Computational Science* team.

nature portfolio

Corresponding author(s): Quanming Yao

Last updated by author(s): Oct 25, 2023

Reporting Summary

Nature Portfolio wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Portfolio policies, see our Editorial Policies and the Editorial Policy Checklist.

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a	Confirmed
☐	☒ The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement
☒	☐ A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
☐	☒ The statistical test(s) used AND whether they are one- or two-sided *Only common tests should be described solely by name; describe more complex techniques in the Methods section.*
☒	☐ A description of all covariates tested
☒	☐ A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
☐	☒ A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
☐	☒ For null hypothesis testing, the test statistic (e.g. *F*, *t*, *r*) with confidence intervals, effect sizes, degrees of freedom and *P* value noted *Give P values as exact values whenever suitable.*
☒	☐ For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
☒	☐ For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
☐	☒ Estimates of effect sizes (e.g. Cohen's *d*, Pearson's *r*), indicating how they were calculated

Our web collection on statistics for biologists contains articles on many of the points above.

Software and code

Policy information about availability of computer code

Data collection	NA
Data analysis	All model code is available at EmerGNN (v 1.0.0) https://github.com/LARS-research/EmerGNN. Other packages used include sklearn (v0.24.2) (https://scikit-learn.org/stable/whats_new/v0.24.html#version-0-24-0), matplotlib (v3.1.3) (https://github.com/matplotlib/matplotlib).

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Portfolio guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A description of any restrictions on data availability
- For clinical datasets or third party data, please ensure that the statement adheres to our policy

Raw data underlying the figures have been included as Source Data.

The data used for experiments is public available in Zenodo at https://doi.org/10.5281/zenodo.10016715. The source code is available in Github at https://github.com/LARS-research/EmerGNN.

There are no figures with associated raw data.
There are no restrictions on data availability.

Human research participants

Policy information about studies involving human research participants and Sex and Gender in Research.

Reporting on sex and gender	n/a
Population characteristics	n/a
Recruitment	n/a
Ethics oversight	n/a

Note that full information on the approval of the study protocol must also be provided in the manuscript.

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences ☐ Behavioural & social sciences ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	In case of experimental data, we used full published datasets (SumGNN in Yu et al, 2021). Hence, the sample size was defined by this study. Specifically, DrugBank dataset contains 1710 drugs, 86 interaction types and 192284 edges, and TWOSIDES dataset contains 604 drugs, 200 interaction types and 252111 edges. In addition, we use HetioNet (Himmelstein et. al. 2017) as the biomedical network, which contains 34124 biomedical entities, 23 relation types and 1690693 edges. For the S1 and S2 settings, we resplit the original data for training, validation and testing with the ratio of 7:1:2 as many related studies did. For each data, we provide 5 different split with 5 different random seeds. More details are provided in Method section.
Data exclusions	No data was excluded from the public datasets listed above.
Replication	Most experiments were performed at least 5 times, rendering very similar results. The experiments are repeatable using the code provided.
Randomization	The samples are split into three fixed sets of training, validation and testing data. All the methods are compared fairly under these datasets.
Blinding	This study does not contain animal experiment, thus do not have concern on blinding.

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

n/a	Involved in the study
☒	☐ Antibodies
☒	☐ Eukaryotic cell lines
☒	☐ Palaeontology and archaeology
☒	☐ Animals and other organisms
☒	☐ Clinical data
☒	☐ Dual use research of concern

Methods

n/a	Involved in the study
☒	☐ ChIP-seq
☒	☐ Flow cytometry
☒	☐ MRI-based neuroimaging