2021 43rd Annual International Conference of the
IEEE Engineering in Medicine & Biology Society (EMBC)
Oct 31 - Nov 4, 2021. Virtual Conference

Predicting Synthetic Lethality in Human Cancers via Multi-Graph Ensemble Neural Network

Mincai Lai[1], Guangyao Chen[1], Haochen Yang[1], Jingkang Yang[2], Zhihao Jiang[1], Min Wu[3] and Jie Zheng[1,4,*]

Abstract— **Synthetic lethality (SL) is currently one of the most effective methods to identify new drugs for cancer treatment. It means that simultaneous inactivation target of two non-lethal genes will cause cell death, but loss of either will not. However, detecting SL pair is challenging due to the experimental costs. Artificial intelligence (AI) is a low-cost way to predict the potential SL relation between two genes. In this paper, a new Multi-Graph Ensemble (MGE) network structure combining graph neural network and existing knowledge about genes is proposed to predict SL pairs, which integrates the embedding of each feature with different neural networks to predict if a pair of genes have SL relation. It has a higher prediction performance compared with existing SL prediction methods. Also, with the integration of other biological knowledge, it has the potential of interpretability.**

I. INTRODUCTION

A. Background

Cancer poses a huge threat to human health. If a new effective cancer drug can be found, it will not only bring considerable economic benefits to pharmaceutical companies but also give cancer patients more chance to survive.

At present, synthetic lethality (SL) is an important way to find new drug targets for cancer treatment. It means that simultaneous inactivation of two non-lethal genes will cause cell death, but loss of either will not.

Finding an SL pair via traditional experimental way needs a lot of resources. Artificial intelligence (AI) can predict possible SL pairs at a much lower cost. Using AI, researchers just need to do experiments to verify those predicted with higher confidence of success than most gene pairs SL, thereby saving considerable costs.

However, there is still little information about known SL pairs, leading to data sparsity. SynLethDB [1] is currently the most comprehensive SL database, but it only contains 36,746 human SL pairs. In fact, there are about 25,000 genes in the human genome, resulting in more than 300 million gene pairs.

B. Existing Studies

Several attempts have been implemented to predict potential SL pairs using AI methods. DiscoverSL [2], SLant [3] and SLRF [4] use random forests to predict SL by analyzing the attributes of genes and the relations between genes (such as PPI, KEGG, etc.). These traditional machine learning methods need much domain expertise and human intervention to extract features.

Matrix decomposition methods, such as GRSM [5] and SL^{2}MF [6], consider the interactions between genes in the training process, but the matrix used for analysis is big and sparse, leading to high analysis cost and thus affecting accuracy.

DDGCN [7] is a graph convolutional network which solved the problem of matrix sparsity, but it does not combine data with knowledge. This method is the first to do SL prediction with Graph Neural Network(GNN). Their work shows that GNN can capture the relations between genes well. It would give biological experts more intuitive comprehension if one method can integrate features about inter-gene relations in the process.

C. Contributions

Here we first enhance the database by deleting gene pairs with errors and adding seven additional features based on other gene relation databases such as GO, Corum, Reactome, etc. which indicate the biological connection of gene pairs.

We propose an novel Multi-Graph ensemble network model which integrates an SL graph and multi gene relation graphs to improve the performance of SL prediction.

II. DATA ACQUISITION

Synthetic lethality data was downloaded from SynLethDB[1] which is the most comprehensive synthetic lethality database. Human SL data in SynLethDB were chosen, as we aim to find drug treatment for human cancers. Some genes in SynLethDB were found to have no corresponding ensemblID or uniprotID. Thus, these genes were excluded from our final dataset. After the pre-processing, we obtained 35,911 SL pairs among 9,862 genes.

To enhance the existing dataset, we added data from other databases related to relations between proteins. We chose five databases, namely Gene Ontology [8], Corum [9], Reactome [10], KEGG [11] and STRING [12] to use. Corum and STRING can provide data of protein complexes and protein-protein interactions(PPIs) respectively. Reactome, KEGG and biological process terms of GO (GO_P) add information

[1] Mincai Lai, Guangyao Chen, Haochen Yang, Zhihao Jiang, Jie Zheng are with School of Information Science and Technology, ShanghaiTech University, China. Email: {laimc, chengy2, yanghch, jiangzhh, zhengjie}@shanghaitech.edu.cn

[2] Jingkang Yang is with School of Life Science and Technology, ShanghaiTech University, China. Email:yangjk@shanghaitech.edu.cn

[3] Min Wu is with the Institute for Infocomm Research, A STAR, Singpore. Email:wumin@i2r.a-star.edu.sg

[4] Jie Zheng is also with Shanghai Engineering Research Center of Intelligent Vision and Imaging, ShanghaiTech University, China.

*To whom correspondence should be addressed.

978-1-7281-1178-0/21/$31.00 ©2021 IEEE

about pathways to the database. The other two terms of GO, molecular function (GO_F) and cellular component (GO_C), were used to add similar functions or locations of genes. Finally, seven types of inter-gene relations were extracted from these databases.

III. Methods

In this section, we first introduce the notations and problem statement, and then present details of our proposed multi-graph ensemble method, including different designs of network structure.

A. Preliminary

Consistent with general GNN methods, our SL graph is denoted by $\mathcal{G}_{sl} = (\mathcal{U}_{sl}, \mathcal{E}_{sl})$, where the nodes $\mathcal{U}$ represent genes, and the edges $\mathcal{E}$ represent SL relations. As the authors of DDGCN [7] defined, SL prediction is formally, given known SL pairs $\mathcal{O}^{+}$, to predict the probability of two gene being an SL pair for each unknown gene pair from $\mathcal{O}^{-}$.

B. Encoder Networks

Besides SL, other relations between genes are used to form different graphs providing diverse biological meanings. The graphs introduced above are denoted as $\mathcal{G}_{corum}$, $\mathcal{G}_{reactome}$, $\mathcal{G}_{kegg}$, $\mathcal{G}_{ppi}$, $\mathcal{G}_{go\_F}$, $\mathcal{G}_{go\_C}$ and $\mathcal{G}_{go\_P}$.

The following steps combine all the gene relation information to do SL prediction. Firstly, we use graph convolutional network (GCN) [13] to generate multi-type node embeddings over our multi-graph. Then these embeddings are introduced to different network models. The GCN is used to learn representations of the nodes by aggregating representations of their immediate neighbours.

A two-layer GCN was implemented to capture node embeddings. The layers are formulated as follows:

$$\mathbf{H}_{\mathbf{A}}^{(1)} = \mathrm{ReLU}\left(\widetilde{\mathbf{A}}\mathbf{X}\mathbf{W}^{(1)} + \mathbf{B},\right)$$
$$\mathbf{H}_{\mathbf{A}}^{(2)} = \widetilde{\mathbf{A}}\mathbf{H}_{\mathbf{A}}^{(1)}\mathbf{W}^{(2)} + \mathbf{B}.$$

In detail, $\widetilde{\mathbf{A}} = \mathbf{D}^{-\frac{1}{2}}\widehat{\mathbf{A}}\mathbf{D}^{-\frac{1}{2}}$ where $\widehat{\mathbf{A}} = \mathbf{A} + \mathbf{I}$ and $\mathbf{D}$ is a diagonal matrix with $\mathbf{D_{ii}} = \sum_{j=1}^{n} \mathbf{A_{ij}}$. $\mathbf{A}$ is the adjacent matrix of gene relation graph. Feature matrix $\mathbf{X}$ is the initial node features. $\mathbf{W}^{(1)}$is the weight matrix for the first GCN layer, $\mathbf{H}_{\mathbf{A}}^{(1)}$ is the output of first GCN layer. The Rectified Linear Unit (ReLU) is the most commonly used activation function in deep learning. The function returns the origin value if it receives a positive input, and otherwise it returns 0.

C. Multi-Graph Ensemble Knowledge Pretrained Network

$\mathcal{G}_{corum}$ was dropped since it is too sparse. The Encoder network includes two GCN layers applied in the following graphs $\mathcal{G}_{ppi}$, $\mathcal{G}_{reactome}$, $\mathcal{G}_{go_c}$, $\mathcal{G}_{go_p}$, $\mathcal{G}_{go_f}$ and $\mathcal{G}_{kegg}$. It was used to capture multi-type node embeddings. Then the sum value of these embeddings was used as input features $\mathbf{X}$ for $\mathcal{G}_{sl}$. The formulation is:

$$\mathbf{X}_{sl} = \mathcal{E}_{ppi} + \mathcal{E}_{reactome} + \mathcal{E}_{go_c} + \mathcal{E}_{go_p} + \mathcal{E}_{go_f} + \mathcal{E}_{kegg}$$

After applying the GCN encoder to generate node embeddings of $\mathcal{G}_{sl}$, for every pair of genes, we take the product of two node embeddings and apply sigmoid to it. The output value can be seen as the predicted probability of two gene being an SL pair. The network structure is shown in Fig.1(a).

D. Multi-Graph Ensemble Fully Connected Network

Seven GCN encoders can capture multi-type node embeddings in parallel. The concatenated embedding was used as input of Fully Connected (FC) Network. The FC network includes three linear layers. This first method is named Multi-Graph Ensemble FC (concat). The network structure is shown in Fig.1(b).

Besides, we also tried two other different method for combination of node embedding. One is summing multi-type node embeddings as one embedding, where is used as input in the FC network. This method is called Multi-Graph Ensemble FC (sum). The other is using seven linear layers as a weighted function, and then generating a weight value of origin embedding. Take $\mathcal{G}_{sl}$ as an example, it can be formulated as follows:

$$\mathcal{E}_{sl} = \mathcal{F}\left(\mathcal{E}_{sl}\right)\mathcal{E}_{sl}$$

Where $\mathcal{F}$ is a linear layer. This method is named Multi-Graph Ensemble FC (weighted concat).

E. Multi-Graph Ensemble Convolutional Neural Network

In order to take full advantage of Convolutional Neural Network (CNN), we adjusted the input matrix. The network structure is shown in Fig.1(c). As this figure shows, each embedding of the another gene relation is adjacent to an SL embedding. Thus, the convolutional kernel is an efficient way to extract relation feature.

For our task, each gene pair has two embeddings, so this matrix has 2 channels. As such, the shape of input matrix is $(14, dim\_embedding, 2)$.

IV. Experimental Results

A. Experiment Setup

1) Dataset Construction: The dataset used in our experiments is constructed from $\mathcal{G}_{sl}$, $\mathcal{G}_{corum}$, $\mathcal{G}_{reactome}$, $\mathcal{G}_{kegg}$, $\mathcal{G}_{ppi}$, $\mathcal{G}_{go\_F}$, $\mathcal{G}_{go\_C}$ and $\mathcal{G}_{go\_P}$. The details are shown in Table I.

TABLE I
The Details of Dataset

Data Category	Number
Gene nodes	9872
$\mathcal{G}_{sl}$ edges	35906
$\mathcal{G}_{corum}$ edges	265
$\mathcal{G}_{reactome}$ edges	11459
$\mathcal{G}_{kegg}$ edges	7308
$\mathcal{G}_{ppi}$ edges	9649
$\mathcal{G}_{go_F}$ edges	28972
$\mathcal{G}_{go_C}$ edges	24187
$\mathcal{G}_{go_P}$ edges	6945

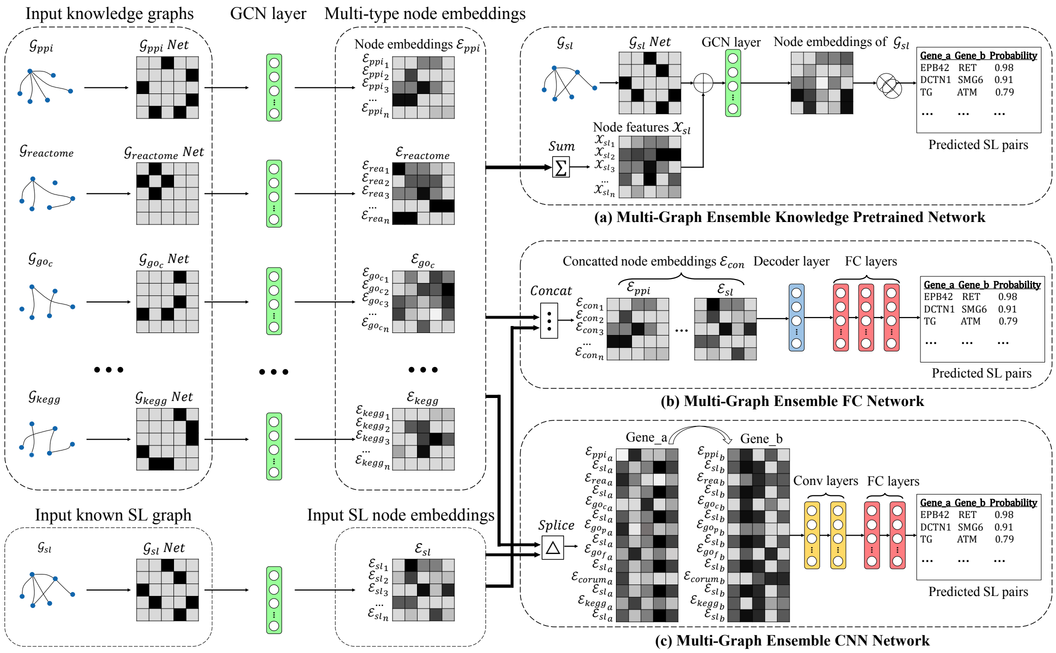

Fig. 1. Multi-Graph Ensemble Network for SL Prediction

2) Dataset Splitting: The main task is to predict the potential SL relation. Therefore, we only split the edges in $\mathcal{G}_{sl}$. Note that all edges in $\mathcal{G}_{sl}$ are positive. Some unknown gene pairs were randomly selected and used as negative. The ratio of our total samples is 1:1 (positive:negative). In order to evaluate the performance of different models, 5-fold cross-validation was used to train our model.

In order to evaluate the performance, 5-fold cross-validation was employed. Noted that we renew the negative samples by random sampling with replacement for each new iteration. This allows the model to be fed more negative samples, making the evaluation more objective.

B. Baselines

We compared our method with some classical GNN models:

- DeepWalk [14]: DeepWalk is a method which learns the node embedding by combines random walk with skip-gram language model.
- node2vec [15]: node2vec is very similar to DeepWalk, it use flexible, biased random walks that can trade off between local and global views of the network.
- SVD [16]: Among SVD is popular for biomedical graph embedding. It focuses on factorizing the first-order adjacency matrix.
- GAE [17]: GAE learn node embeddings by a GCN encoder and an inner product decoder.
- GraphSAGE [18]: GraphSAGE can be viewed as a stochastic generalization of graph convolutions, it learn how to propagate information across the graph to compute node features.

We reproduced some methods for SL prediction:

- SL^{2}MF [6]: SL^{2}MF is factorization-based methods for SL prediction, which learns the embedding of genes by logistic matrix factorization.The GO annotations and the topological features of the PPI network are employed for calculate the similarity.
- DDGCN [7]: DDGCN is at present one of the state-of-the art models in SL prediction. It propose a novel Dual-Dropout GCN structure for learning more robust gene representations. DDGCN addresses the overfitting problem on sparse graphs by employing both coarse-grained node dropout and fine-grained edge dropout.

C. Multi-Graph Ensemble

In our experiment, the numbers of neurons in the two GCN layers are 128 and 16. We adopted exponential learning rating, meaning this model will decay the learning rate of each parameter group by γ in every epoch. The initial leaning rate was 0.01. In Multi-Graph Ensemble CNN, the size of the convolution kernel was 2 and the dropout rate was 0.1.

D. Results of Performance Comparison

With the above experiment settings, the results are shown in Table II.

1) Comparison with Baselines: For SL prediction, MGEs achieved top performance in terms of all the three metrics. While DDGCN was considered the champion of SL prediction, MGE FC (concat) is able to achieve further improvement by 10.9% on AUROC metric and 5.8% on AUPR metric. And the F1 metric increased 6.5% on MGE CNN. At the same time, compared with other methods, MGEs can increase about 10% on the three metrics. The substantial performance improvement of MGEs was likely due to importing external knowledge and the idea of multi-graph ensemble.

Since we train seven kinds of node representations at the same time, the number of parameters of node embedding increased to seven times of baselines. Hence our training for node embedding end-to-end has increased computational complexity. But the inference time does not change significantly since our classification network is lightweight.

2) Comparison among Model Variants: Among the four model variants of MGE in Table II, MGE FC (concat) showed stength in AUROC and AUPR, and is competitive in F1. We think this is because the concat method retains more knowledge information and a simpler FC network is more effective than a complex CNN network.

TABLE II

Performance comparison of various methods

	AUROC	AUPR	F1
DeepWalk	0.8492	0.8697	0.7872
node2vec	0.8388	0.8424	0.7673
SVD	0.8627	0.8795	0.7972
GAE	0.7300	0.7340	0.6760
GraphSAGE	0.7458	0.8364	0.6971
SL^{2}MF	0.7734	0.8589	0.7330
DDGCN	0.8460	0.8977	0.8140
MGE Pretrain	0.9465	0.9512	0.5518
MGE FC (sum)	0.9381	0.9432	0.8631
MGE FC (concat)	**0.9553**	**0.9555**	0.8783
MGE FC (weighted concat)	0.9381	0.9432	0.8631
MGE CNN	0.9521	0.9516	**0.8786**

E. Prediction of New SL Pairs

We predicted an unknown SL pair from the existing data, TBK1 and VHL, with the probability of 0.93, later we found it had been experimentally discovered by Hu et al.[19] in May 2020. TBK1 and VHL appeared only 45 times and once in the original SynLethDB respectively. Also we found that 65 genes apperaed more than 100 times in SynLethDB. In other words, we can accurately predict the SL relation of genes with fewer occurrences, showing that our prediction is valuable.

V. Conclusion and Future Work

In this paper, we proposed a framework of multi-graph ensemble network, and assembled MGE knowledge pretrained network, MGE FC network and MGE CNN, which improved the performance of SL prediction. Also, the original SL database was enhanced.

In the future, the interpretability of the network will be implemented based on attention mechanism. In addition, since we constructed multiple gene-gene interation graphs, we can take the link prediction task on PPI, GO_C, GO_P and so on. Then the SL prediction task can be integrated into a multi-task learning framework. The link prediction in a knowledge graph can be considered as an auxiliary task. The execution of these auxiliary tasks may improve the prediction performance of our main task of SL prediction.

References

[1] J. Guo, H. Liu, and J. Zheng, "Synlethdb: synthetic lethality database toward discovery of selective and sensitive anticancer drug targets," *Nucleic Acids Research*, vol. 44, no. D1, pp. D1011–D1017, 2016.

[2] S. Das, X. Deng, K. Camphausen, and U. Shankavaram, "Discover sl: an r package for multi-omic data driven prediction of synthetic lethality in cancers," *Bioinformatics*, vol. 35, no. 4, pp. 701–702, 2019.

[3] G. Benstead-Hume, X. Chen, S. R. Hopkins, K. A. Lane, J. A. Downs, and F. M. Pearl, "Predicting synthetic lethal interactions using conserved patterns in protein interaction networks," *PLoS Computational Biology*, vol. 15, no. 4, p. e1006888, 2019.

[4] J. Li, L. Lu, Y.-H. Zhang, M. Liu, L. Chen, T. Huang, and Y.-D. Cai, "Identification of synthetic lethality based on a functional network by using machine learning algorithms," *Journal of Cellular Biochemistry*, vol. 120, no. 1, pp. 405–416, 2019.

[5] J. Huang, M. Wu, F. Lu, L. Ou-Yang, and Z. Zhu, "Predicting synthetic lethal interactions in human cancers using graph regularized self-representative matrix factorization," *BMC Bioinformatics*, vol. 20, no. 19, pp. 1–8, 2019.

[6] Y. Liu, M. Wu, C. Liu, X. Li, and J. Zheng, "Sl^{2}mf: Predicting synthetic lethality in human cancers via logistic matrix factorization," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2019.

[7] R. Cai, X. Chen, Y. Fang, M. Wu, and Y. Hao, "Dual-dropout graph convolutional network for predicting synthetic lethality in human cancers," *Bioinformatics*, 2020.

[8] The Gene Ontology Consortium, "The Gene Ontology Resource: 20 years and still GOing strong," *Nucleic Acids Research*, vol. 47, no. D1, pp. D330–D338, 11 2018.

[9] M. Giurgiu, J. Reinhard, B. Brauner, I. Dunger-Kaltenbach, G. Fobo, G. Frishman, C. Montrone, and A. Ruepp, "CORUM: the comprehensive resource of mammalian protein complexes—2019," *Nucleic Acids Research*, vol. 47, no. D1, pp. D559–D563, 10 2018.

[10] B. Jassal, L. Matthews, G. Viteri, C. Gong, P. Lorente, A. Fabregat, K. Sidiropoulos, J. Cook, M. Gillespie, R. Haw, F. Loney, B. May, M. Milacic, K. Rothfels, C. Sevilla, V. Shamovsky, S. Shorser, T. Varusai, J. Weiser, G. Wu, L. Stein, H. Hermjakob, and P. D'Eustachio, "The reactome pathway knowledgebase," *Nucleic Acids Research*, vol. 48, no. D1, pp. D498–D503, 11 2019.

[11] M. Kanehisa, M. Furumichi, M. Tanabe, Y. Sato, and K. Morishima, "KEGG: new perspectives on genomes, pathways, diseases and drugs," *Nucleic Acids Research*, vol. 45, no. D1, pp. D353–D361, 11 2016.

[12] D. Szklarczyk, A. L. Gable, D. Lyon, A. Junge, S. Wyder, J. Huerta-Cepas, M. Simonovic, N. T. Doncheva, J. H. Morris, P. Bork, L. J. Jensen, and C. Mering, "STRING v11: protein–protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets," *Nucleic Acids Research*, vol. 47, no. D1, pp. D607–D613, 11 2018.

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

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

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

[16] L. Zhu, Z.-H. You, and D.-S. Huang, "Increasing the reliability of protein–protein interaction networks via non-convex semantic embedding," *Neurocomputing*, vol. 121, pp. 99–107, 2013.

[17] T. N. Kipf and M. Welling, "Variational graph auto-encoders," *arXiv preprint arXiv:1611.07308*, 2016.

[18] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Advances in Neural Information Processing Systems*, 2017, pp. 1024–1034.

[19] L. Hu, H. Xie, X. Liu, F. Potjewyd, L. I. James, E. M. Wilkerson, L. E. Herring, L. Xie, X. Chen, J. C. Cabrera, *et al.*, "Tbk1 is a synthetic lethal target in cancer with vhl loss," *Cancer Discovery*, vol. 10, no. 3, pp. 460–475, 2020.