

Article

Interpreting the Mechanism of Synergism for Drug Combinations Using Attention-Based Hierarchical Graph Pooling

Zehao Dong ¹, Heming Zhang ², Yixin Chen ¹, Philip R. O. Payne ² and Fuhai Li ^{2,3,*}

¹ Department of Computer Science & Engineering, Washington University in St. Louis, St. Louis, MO 63130, USA; zehao.dong@wustl.edu (Z.D.); chen@cse.wustl.edu (Y.C.)

² Institute for Informatics, Data Science, and Biostatistics, Washington University School of Medicine, Washington University in St. Louis, St. Louis, MO 63110, USA; hemingzhang@wustl.edu (H.Z.); prpayne@wustl.edu (P.R.O.P.)

³ Department of Pediatrics, Washington University School of Medicine, Washington University in St. Louis, St. Louis, MO 63110, USA

* Correspondence: fuhai.li@wustl.edu

Simple Summary: This paper introduces a novel graph neural network (a hierarchical graph pooling model), SANEpool, to effectively detect core sub-networks of significant genes for predicting the synergy score of drug/drug combinations in cancer. SANEpool successfully addresses the limitations of the un-transparency in the prediction process of previous computational AI models for drug synergy prediction, while providing the superior predictive performance than popular baselines on numerous drug-synergy prediction datasets. The success of SANEpool indicates that significant gene-gene interactions and gene-drug interactions play a crucial role in designing powerful deep learning models to provide accurate prediction and to reveal the mechanism of the synergy (MoS).

Abstract: Synergistic drug combinations provide huge potentials to enhance therapeutic efficacy and to reduce adverse reactions. However, effective and synergistic drug combination prediction remains an open question because of the unknown causal disease signaling pathways. Though various deep learning (AI) models have been proposed to quantitatively predict the synergism of drug combinations, the major limitation of existing deep learning methods is that they are inherently not interpretable, which makes the conclusions of AI models untransparent to human experts, henceforth limiting the robustness of the model conclusion and the implementation ability of these models in real-world human–AI healthcare. In this paper, we develop an interpretable graph neural network (GNN) that reveals the underlying essential therapeutic targets and the mechanism of the synergy (MoS) by mining the sub-molecular network of great importance. The key point of the interpretable GNN prediction model is a novel graph pooling layer, a self-attention-based node and edge pool (henceforth SANEpool), that can compute the attention score (importance) of genes and connections based on the genomic features and topology. As such, the proposed GNN model provides a systematic way to predict and interpret the drug combination synergism based on the detected crucial sub-molecular network. Experiments on various well-adopted drug-synergy-prediction datasets demonstrate that (1) the SANEpool model has superior predictive ability to generate accurate synergy score prediction, and (2) the sub-molecular networks detected by the SANEpool are self-explainable and salient for identifying synergistic drug combinations.

Keywords: drug response prediction; graph neural networks; interpretability



Citation: Dong, Z.; Zhang, H.; Chen, Y.; Payne, P.R.O.; Li, F. Interpreting the Mechanism of Synergism for Drug Combinations Using Attention-Based Hierarchical Graph Pooling. *Cancers* **2023**, *15*, 4210. <https://doi.org/10.3390/cancers15174210>

Academic Editors: Rick L. Stevens and Yitan Zhu

Received: 9 June 2023

Revised: 16 August 2023

Accepted: 17 August 2023

Published: 22 August 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Combinatorial drug therapy has been of crucial importance in modern clinical disease treatment and drug discovery [1]. Synergistic drug combination can produce more beneficial combinatorial effects than each constituent, and the synergic behavior always allows the lower doses of the drugs in the combination relative to their individual potencies,

thus reducing the induction of drug resistance [2] and overcoming the side effects [3–5] associated with the high doses of single drug usage. Hence, drug combination therapy provides a greatly promising avenue towards the treatment of the most dreadful multi-factorial diseases, such as cancer [6–9], diabetes, and bacterial infections. In contrast to the synergism, the therapeutic efficacy of some drug combinations can be simply additive or even sub-additive. As such, there has been growing interest in investigating the synergy mechanism of drug combinations to distinguish the synergistic combinations from non-synergistic ones.

Frequently, the synergy of drug combinations is tested in pre-clinical model environments, such as high-throughput screening (HTS) instruments [10,11], where thousands of combinatorial experiments are simultaneously implemented under actionable hypotheses and conditions to profile the synergism. However, the testing space can be extremely massive due to the large amount of drug combinations, cell lines, dose choices, and patient samples, hence it can be impractical to traverse the whole testing space [12]. Furthermore, the transition from some pre-clinical environments to the clinical practice sometimes can also cause failure [13]. As such, various computational (AI) models [14–17] are developed to assist the synergy analysis of drug combinations.

Most computational AI models take massive omics data and chemical structure data as input and then adopt deep learning algorithms to predict the synergy score to determine the presence of the synergism. Several machine learning models, such as TreeCombo [16] and random forest [18], build ensemble trees to predict the synergy scores and have achieved impressive results. After that, numerous deep learning models were proposed in the domain to unleash the predictive power of neural networks. A large body of work, including DeepSynergy [17], MatchMaker [19], and CCSynergy [20], shows that wisely combining the drug profiles and gene expression profiles in specific cell lines as input features enables the vanilla multiple layer perceptron (MLP) to accurately predict the synergy scores of drug combinations, while TranSynergy [21] applies attention-based transformer architecture to boost the prediction performance. On the other hand, SDCNet [22] and DeepDDS [23] demonstrate that modeling the connections between drugs and genes can benefit synergy prediction and propose to encode the networks/graphs that consist of genes and drugs through graph neural networks (GNNs).

In addition to the predictive ability, the interpretability of deep learning models is desirable in real-world scenarios like the pharmacy industry and healthcare, as it allows us to incorporate human expertise in decision making to provide more robust conclusions. Currently, limited existing works provide interpretable predictions of drug synergy. TranSynergy [21] applies the post hoc interpretation framework [24] that computes the Shapley value of each gene through GradientExplainer and uses DeepExplainer to characterize its contribution to the final synergy prediction. Though post hoc interpretation mechanisms that produce interpretations after the model creation work well in some cases, the ante hoc interpretable model [25] is still missing in the domain of drug synergism analysis to inject interpretability from the beginning of the model design. Consequently, the objective of this paper is to develop deep learning models to generate accurate and interpretable synergic predictions, and we resort to the graph pooling methodology in graph neural networks (GNNs).

In recent years, GNNs have been the dominant architecture for analyzing graph-structured data, such as social networks [26,27], protein networks [28,29], circuit networks [30,31], etc. Most GNNs follow the neighborhood aggregation scheme that updates each node feature by propagating its neighboring node features to its current feature and have achieved impressive results on various graph learning tasks, ranging from node classification [32] and link prediction [33,34] to graph classification [35]. In order to generate a subset of nodes or cluster of nodes for the prediction tasks, several graph-pooling models are proposed. DGCNN [36] proposes to sort nodes for pooling according to their structural roles within the graph. However, since it stacks multiple graph convolution layers to propagate information and then globally implement the graph down-sampling

via a pooling module, the generated graph representation is inherently flat. In order to extract hierarchical graph representations, DiffPool [37] uses different GNNs to separately implement neighborhood aggregation and graph pooling, and it provides a framework to hierarchically pool nodes across a broad set of graphs.

Following this inspiration, we introduce a novel hierarchical graph pooling model, SANEpool (self-attention-based node and edge pool), for the interpretable drug synergism prediction task, which aims to reveal the drug combination synergism by systematically extracting the target gene sub-network that intrigues the synergic behavior. Various medical chemistry research has shown that cancer is driven by genetic and epigenetic alterations, many of which can be mapped into signaling pathways that control the survival and migration/invasion of cancer cells. As one previous signaling pathway analysis suggests [38], 89% of tumor samples had at least one driver alteration in one of ten cancer-related signaling pathways that is responsible for tumor development, while 57% and 30% had one and multiple potentially druggable targets, respectively. Another example [39] is that the drug combination of venetoclax and idasanutlin can generate antileukemic efficacy in the treatment of acute myeloid leukemia by inhibiting antiapoptotic Bcl-2 family proteins and activating the p53 pathway at same time. Thus, inhibited signaling targets analysis shows great potential of facilitating drug combination synergism discovery. Following this intuition, each SANEpool layer implements the standard graph convolution layer (GCN) to generate attention features that encode the gene (and drug) information as well as the topology information of the molecular network. Based on these attention features, the probability that a gene or an interaction (gene–gene interaction, gene–drug interaction) will cause the synergy performance is calculated, and then genes and interactions that are unlikely to influence the synergism of drug combination will be filtered out. The proposed model is composed of multiple SANEpool layers and will output the target gene sub-network for interpretable and robust synergy prediction.

We evaluate our SANEpool model on three popular drug-synergy-prediction datasets, which are constructed upon NCI ALMANAC [18], GDSC (Genomics-of-Drug-Sensitivity-in-Cancer) [40], and O’Neil [5] experimental settings. Experimental results demonstrate that the SANEpool model achieves the current state-of-the-art performance for all datasets. Furthermore, through visualizations of the detected target gene sub-network of different cancer cell lines, we observe that the proposed model (SANEpool) can detect the salient target gene patterns that cause the synergic drug combinations, which reveals the synergism mechanism in drug combination discovery.

2. Other Related Work

2.1. Graph Neural Networks

Graph neural networks (GNNs) have revolutionized the field of learning with graph-structured data and empirically achieved the current state-of-the-art performance in various graph learning tasks, ranging from node classification and link prediction to graph classification. Broadly, GNNs [35–37,41–43] follow a recursive neighborhood aggregation scheme where the node features from the neighborhood of each node are aggregated to update the node’s feature. Such frameworks allows GNNs to capture the graph topology as well as node features, hence unleashing the representation learning ability among graphs.

2.2. Pan-cancer Biomarkers

The genotype-oriented therapies for pan-cancer biomarkers have been approved by the US Food and Drug Administration. These biomarkers amplify our knowledge of genomic profiling across various malignancies by revealing the prevalence of certain oncogenic alternations, hence playing important roles in drug combination discovery. According to a previous study [44], 30% of recurrent alternations across tumor types from 10,000 patients with metastatic cancers are targetable, and various genotype-oriented therapies are detected based on genomic profiling. For instance, the neurotrophic receptor kinase (NTRK) family genes 1–3 were identified in various pediatric cancers. Then, a clinical trial of the Trk

inhibitor larotrectinib demonstrated the antitumor activity and hence led to the usage of arotrectinib as treatment for cancers harboring NTRK fusions.

2.3. Machine Learning in Drug Synergy Prediction

The drug synergy analysis is beneficial as it provides a useful resource for novel predicted drug combinations. However, manually discovering the synergism in practice is still challenging due to the high cost and the limited number of synergistic drug combinations approved by the Food and Drug Administration. Hence, the computational model shows huge potential to find the mechanism of synergy (MoS) in a biologically meaningful manner. Currently, various computational models, ranging from unsupervised learning models [45–47] to supervised learning models [48,49], have been proposed for the purpose of predicting the synergy of drug combinations and have achieved expressive performance. Broadly, these computational models, such as DeepSynergy [17] and Matchmaker [19], take as input massive chemical descriptors of tested drug pairs and cell-line gene expression profiles and then use multi-layer-perceptron (MLP)-based deep learning models to predict the synergy score of drug combinations. Although these models effectively predict the synergy score of drug combinations, they are inherently not interpretable, while the interpretability is crucial for the real-world application. As the drug synergy has been reported to be largely determined by the biomolecular network topology [50], many deep learning models, such as DeepSignalingSynergy [14] and IDSP [51], incorporate the gene–gene interactions and gene–drug interactions into model design to allow the model make interpretable predictions that explain the underlying MoS.

3. Methodology

In this section, we introduce the proposed graph pooling model, self-attention-based node and edge Pool (SANEpool), on the basis of which we develop an interpretable graph neural network to detect the biologically meaningful gene sub-network for synergic and interpretable drug combination prediction. The key point of SANEpool is to contain the attention score (importance) of nodes and edges through the node features and the graph topology; then, the attention scores make it possible to filter out less important (less relevant) nodes and edges for prediction. In Section 3.1, we introduce the problem formulation of the interpretable drug prediction task. In Section 3.2, we develop the mechanism of SANEpool, and the overall interpretable model architecture is described in Section 3.3. The problem formulation and the architecture of SANEpool are illustrated in Figures 1 and 2, respectively.

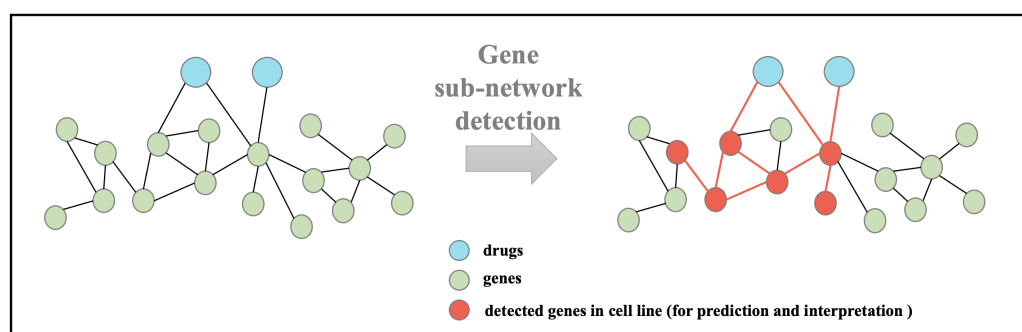


Figure 1. Overview of the problem formulation. The objective is to systematically detect gene sub-networks to predict the synergy score of the input drug pairs and to explain the drug combinatorial synergies across a large group of drug combinations and cell lines.

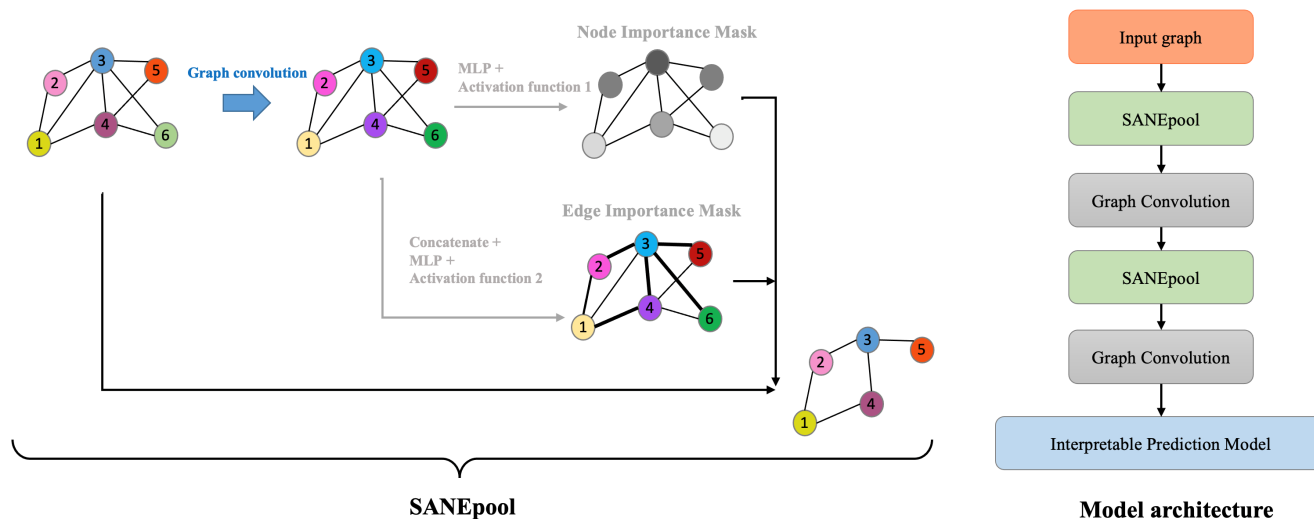


Figure 2. The proposed SANEpool layer and the overall architecture. SANEpool layer incorporates node features and graph topologies through a GNN layer, and then the output is used to compute the attention score of nodes and edges, on the basis of which we can detect important nodes and edges through top- k sorting or threshold filtering. The overall model takes a hierarchical graph pooling architecture.

3.1. Problem Configuration

In this work, we study the molecular networks of cancer drug combination therapies in an inductive manner, where the tested drug pairs are unseen during the training phase. The molecular networks contain drugs and genes in the signaling pathways. The objective is to predict the synergic score of each drug pair based on the molecular network. In order to make the prediction interpretable, SANEpool is proposed to detect the sub-gene network (i.e., the red gene nodes in Figure 1 which consist of a subset of genes in signaling pathways that are most relevant to the synergistic effect of the drug pair). Then, the detected sub-gene network provides insight into the molecular mechanism of resistant or sensitive responses to cancer drug combinations.

Let $G = (V, E)$ be the molecular network (graph), where V is the node set that contains gene nodes and drug nodes, E is the edge set that characterizes the interactions between nodes. For the notation convenience, we use A to denote the adjacency matrix of the graph. Since the molecular graph is inherently undirected and has no self-loop, adjacency matrix A is a symmetric matrix. We use $X \in R^{n \times h}$ to denote the input node feature, where n is the number of nodes, and h is the dimension of input features. Hence, the graph can also be represented as the pair of the node feature and adjacency matrix such that $G = (X, A)$. Furthermore, we use $Z^t \in R^{n \times h_t}$ to denote the node representation in layer t , where h_t is the dimension of node representation.

3.2. The Proposed SANEpool Model

The (self-)attention mechanism plays important role in various machine learning models, including natural language processing architectures [52–54], graph classification architectures [55], sequential prediction algorithms [56], adversarial learning models [57], etc. The attention mechanism allows input features to be the criteria for the attention itself [58] and thus can distinguish the relative importance between features during the information aggregation process. In order to incorporate the node features and graph topologies in the attention scores for the nodes and edge pooling, we follow the idea of SAGpool [59] and utilize a graph convolution layer to aggregate such information and to compute the attention feature matrix H^t .

$$H^t = f(\tilde{D}^{-1} \tilde{A} Z^t \Theta^t) \quad (1)$$

where $\tilde{A} = A + I$ is the adjacency matrix with added self-loops, \tilde{D} is the corresponding diagonal degree matrix of \tilde{A} such that $\tilde{D}_{ii} = \sum_{j=1}^n \tilde{A}_{ij}$, and f is the activation function. The

matrix $\Theta^t \in \mathbb{R}^{h_t \times h_{t+1}}$ is the trainable parameter to coordinate the attention score of nodes and edges, where h_t is the feature dimension in layer t . Various graph convolution layers [43,60,61] have been proposed; these GNN formulas can be used as substitution of Equation (1). All GNN layers follows the same information aggregation framework; hence, the extracted attention features H^t contain information of node features as well as graph topologies.

Next, we discuss how to compute the attention scores of nodes and edges based on extracted attention features H^t . The node attention score is determined as the cosine of the angle between the attention feature vector and a trainable projection/parameter vector p^t of size h_t (Equation (2)). **The attention score Att measures the probability that gene nodes will cause the synergism of drug combinations.** Hence, we can sort node attention scores and adopt the top- k selection technique to hierarchically select the gene sub-network for the purpose of synergism prediction, and such a process is formulated as

$$Att_i^t = \frac{(p^t)^T H_i^t}{\|p^t\|^{\frac{1}{2}} \|H_i^t\|^{\frac{1}{2}}} \quad (2)$$

$$idx^t = top(Att^t, k) \quad (3)$$

The core idea of the proposed SANepool is to filter out ‘less important’ genes (and connections), and then only kept genes (and connections) are used to predict the synergism of drug combinations. Equation (2) indicates that the attention score Att is normalized and has a value in the set $[-1, 1]$. Thus, a gene with larger attention score will be more likely to be kept in the top- k selection process (i.e., Equation (3)), indicating that the gene has a higher probability to be selected to predict/interpret the synergism of drug combination.

In the top node selection process (i.e., Equation (3)), when $k \in N$, we adopt the top- k node selection method as DGCNN [36]. On the other hand, we can also implement node selection method proposed by [62] to retain a proportion of nodes when $k \in (0, 1]$. Based on selected node index idx , we can construct the graph downsampling as $\tilde{Z}^{t+1} = Z^t(idx, :)$ and $A^{t+1} = A^t(idx, idx)$.

For general graph learning problems, it can be difficult to find the reasonable pre-defined indexes for nodes in graphs, as the problem is equivalent to the graph isomorphism problem, which is known to be NP-hard. However, for gene networks, since each gene at most appears once in each network, we can universally assign each possible gene (that appears in at least one network/graph in the data set) a unique (pre-defined) index. For instance, we can collect all possible genes and lexicographically sort their names, and then the order of genes in the sorting operation can be used as the index. Consequently, the pre-defined index is equivalent to the gene. It does not matter if we change the pre-defined index system once these indexes can injectively distinguish genes and are consistent among gene networks. The main advantage is to reduce the space complexity. The gene network usually contains lots of genes (i.e., a very large n), then, to store all possible edge pairs in the layer t , we require a tensor \mathbb{T} with the shape of $\mathbb{R}^{n \times n \times 2 \times h_t}$, where h_t is the size of node representation vectors in layer t and usually is selected from the set $\{32, 64, 128, 256, 512\}$. Then an MLP is used to learn the edge attention matrix from \mathbb{T} . In contrast, using pre-defined indexes can reduce the size of \mathbb{T} to $\mathbb{R}^{n \times n}$, and no MLP is needed in this step. Hence, we can significantly shrink the model/algorithm complexity.

Let $G^{t+1} = (Z^{t+1}, A^{t+1})$. Since the proportion of retained information (attention score) of nodes in G^{t+1} are different, the connectivity strength between nodes can be different. Hence, we should also provide a consistent mechanism to characterize such bias. One intuitive approach is to apply the graph attention mechanism based on the extracted attention features H^t :

$$e_{i,j}^t = relu(MLP(H_i^t \parallel H_j^t)) \quad (4)$$

where the symbol \parallel indicates the concatenation operation. Due to the universal approximation theorem [63,64], such formulations can approximate any continuous function that measures the connectivity strength. However, a major limitation of this framework is the memory cost. For large-scale graphs, the memory usage to compute the attention

score of edge may limit the practical ability of the proposed model. Luckily, the molecular network (graph) takes advantage that each gene node in the graph has a corresponding predefined index (i.e., the gene name), which serves as a canonical node order. Hence, we can directly model the interaction strength in each layer t through a trainable parameter matrix W^t . The edge weight in the subgraph G^{t+1} is trainable through the equation $A^{t+1} = A^{t+1} \circ W^t(\text{idx}^t, \text{idx}^t)$, where \circ denotes the Hadamard product operation. The advantage of this formulation is discussed in Appendix D.

3.3. The Overall Architecture

3.3.1. Hierarchical Graph Pooling

The overall architecture of the proposed interpretable model takes the hierarchical graph pooling structure [59,62]. Figure 2 illustrates the overall architecture, and details are provided in Appendix E. The model stacks multiple SANEpools followed by a graph convolution layer to hierarchically extract a key sub-graph from the input graph. In other words, the proposed SANEpools layer is used to downsample the important sub-network (sub-graph). After the downsampling process, another GNN (graph convolution) layer is used to aggregate information based on sub-graph G^{t+1} (Equation (5)) to update the node representation.

$$Z^{t+1} = \text{GNN}(Z^t[\text{idx}^t, :], A^{t+1}) \quad (5)$$

where $Z^t[\text{idx}^t, :]$ is the node representation matrix of the downsampled sub-graph in the t -th SANEpools layer. Then, the output of the last graph convolution layer is used for the prediction task.

3.3.2. Readout Mechanism

Inspired by IGMC [65], the proposed model takes the node representation of two drug nodes to make prediction. The graph convolution framework indicates that such node representations encode the enclosed subtrees around the drug nodes in the pooled graph, hence representing the relations and interactions between drugs and genes. Ideally, we hope that the readout phase should be invariant to the drug node order, as the same drug pairs always have the same clinical performance regardless their order. Let u_1, u_2 be the output of two drug nodes; then, the readout layer follows the factorization decoder in decagon [19]:

$$\text{score} = u_1^T D^T D u_2 \quad (6)$$

where $D \in \mathbb{R}^{t_L \times t_L}$ is the trainable parameter matrix in the decoder, where t_L is the dimension of node representations in the last graph convolution layer L . The parameter matrix D models the interaction effects between every two dimensions in drug representations u_1 and u_2 .

It can be shown that the above readout function is invariant to the order of drugs. First, since the vectors/embeddings of two drugs, u_1 and u_2 , are generated by GNN message passing layers, which are invariant to the order of nodes/vertices in the input graphs, u_1, u_2 will not change if we permute the order of drugs (even the order between two drugs and genes). Second, Equation (6) uses a symmetric function to compute the synergy score based on u_1 and u_2 and thus is permutation-invariant to the order of two drugs.

3.4. Comparison to Related Works

3.4.1. Comparison to Other Graph Pooling Models

Both SANEpools and Sortpools [36] propose to sort nodes according to the structural role (i.e., ‘importance’) of nodes in the graph. However, DGCNN is inherently flat, while SANEpools aggregate information in a hierarchical way. Thus, SANEpools is capable of capturing more informative global features for the downstream prediction task. On the other hand, SANEpools and Diffpools [37] learn graph representation in a hierarchical way. However, Diffpools focuses on the relational analysis of node clusters, while SANEpools

detects the critical sub-network based on the downstream tasks. Hence, SANEpools support the pathway-based analysis in the drug combination prediction, thus providing interpretable results for healthcare.

3.4.2. Comparison to Other GNNs for Drug Synergy Prediction

The proposed SANEpools model, SDCNet [22], and DeepDDS [23] share the same motivation that utilizes GNNs to capture the useful relational information of drugs and genes in the drug synergy prediction. However, they use GNNs to extract different types of relational information. The proposed SANEpools model encodes the important interactions between drugs and target genes in cell lines; SDCNet learns the cell-line-specific drug interactions, while DeepDDS encodes the molecular graph of each drug, which is generated by RDKit [66] based on the drug's chemical structure. Consequently, compared to SANEpools, SDCNet and DeepDDS rely on more drug profile information. Furthermore, SDCNet is used for the simpler classification task which aims to predict the synergistic effects (0/1 classification) instead of accurately predicting the synergy score.

4. Experiments

We evaluate the predictive ability of the proposed SANEpools by comparing the accuracy of the estimated synergy score against popular baselines. Furthermore, to illustrate the synergism detected by SANEpools, we also implement visual analytics and statistical analysis to show that the proposed SANEpools can detect significantly different causal gene sub-networks for synergic drug combinations and non-synergic drug combinations in each cell line.

4.1. Dataset Description

We evaluate the effectiveness of the proposed SANEpools model on three drug-synergy-prediction datasets: NCI-DCD (NCI-Almanac-based Drug Combination Dataset), GDSC-SDD (Genomics-of-Drug-Sensitivity-in-Cancer [40]-based Single Drug Dataset) and O'Neil-DCD (O'Neil [5]-based Drug Combination Dataset). Overall, these datasets take input molecular networks/graphs, which consist of drug combinations/pairs and genes in the target cell lines, where gene expressions are used as node features in the input molecular networks/graphs. The objective is to predict the score/synergy score of each drug/drug pair. In all datasets, the edges/interactions between drugs and genes are collected from the DrugBank database (version 5.1.5, released 3 January 2020) [67], while the edges/interactions between genes are collected from the KEGG (Kyoto Encyclopedia of Genes and Genomes) database [68] based on the physical signaling interactions from documented medical experiments. The synergy score corresponding to each drug pair is computed as the average combo-score [39] with different doses on a given tumor cell line. The difference between datasets is the source of drug combinations and signaling pathways.

4.1.1. NCI-DCD Dataset

The NCI-DCD ensemble genes from 46 well-known signaling pathways (45 "signaling pathways" + cell cycle) [69] in KEGG. Drug combinations are collected from the DrugBank database [67], whose target genes are included in the aforementioned 46 signaling pathways, and the combo-scores of drug combinations are available from the NCI Almanac dataset. We provide details of the 46 signaling pathways and 21 selected FDA-approved drugs in Appendix B. In summary, NCI-DCD contains 5658 graphs/networks. Each graph/network has 1364 genes and two drugs, while containing about 25,000 edges that connect genes and drugs.

4.1.2. O'Neil-DCD Dataset

In O'Neil-DCD, signaling pathway information is also formulated based on the gene expression data of 1047 cancer cell lines in the Broad Institute Cancer Cell Line Encyclopedia (CCLE) database [70]. Drug combinations and their corresponding synergy scores are collected from O'Neil datasets [5], whose target genes are included in KEGG database [68].

In summary, there are in total 4637 graphs/networks, and each contains two drug nodes and 1823 gene nodes.

4.1.3. GDSC-SDD Dataset

In GDSC-SDD, signaling pathway information is formulated based on the gene expression data of 791 cancer cell lines in the Broad Institute Cancer Cell Line Encyclopedia (CCLE) database, while the corresponding drug/cancer-cell-line response data set are available in the Genomics of Drug Sensitivity in Cancer (GDSC) database. In the experiment, there are in total 16,761 graphs/networks, and each contains a drug node and 969 gene nodes. In the dataset, since there is only a single drug node in the input network/graph rather than a pair of drug nodes, we can not use Equation (6) as the readout function. Instead, we use a two-layer MLP that takes as input the learnt embedding of the drug node in this setting.

The three benchmark datasets (i.e., NCI-DCD, O'Neil-DCD, GDSC-SDD) contain different numbers of networks/graphs, where each drug or drug pair on each cell line represents one network/graph. In each network/graph, gene nodes use three features as the input/initial node features: gene expression value and two 0/1 indicators to indicate whether the gene is connected to two drugs, while drug nodes set all these values as -1 . Then, when two networks correspond to different cell lines, the same gene will have different gene expression values as the initial node feature in these two networks. Furthermore, each network contains a gene if and only if (1) its gene expression data is available and (2) the gene is available in the KEGG dataset to track the interactions between genes. Then, since NCI-DCD, O'Neil-DCD, and GDSC-SDD have different resources of cell-line-based gene expression data (NCI-DCD dataset uses gene expression data from its website, and the gene expression data were collected from CCLE for the other two datasets) and drug (pair)/synergy data, networks in the different datasets will contain different sets of genes.

4.2. Baseline Methods

In popular deep learning models for drug-synergy prediction, we select three popular baselines: DeepSynergy [17], DeepSignalingSynergy [14], and TransSynergy [21]. Compared to other methods, DeepSynergy uses additional drug profile features. To make a fair comparison, we mask out additional input features with 0 embeddings.

We also compare SANEpools with six widely adapted GNN baselines. These baselines can be categorized into two types: (1) flat GNNs: Graph Attention Network (GAT) [71], Deep Graph CNN (DGCNN) [36], Graph Isomorphism Network (GIN) [72], and graph convolutional network (GCN) [61] and (2) popular graph pooling models: Diffpool [37] and SAGpool [59]. For GCN, GIN, and GAT, we stack three graph convolution layers with 64 output feature channels and concatenate sum-pooled features from the layers to generate the graph representation, which is then passed to an MLP to predict the graph label.

In the SANEpools model, we keep 200 nodes in the last layer and keep 90% edges in each SANEpools layer. The SANEpools model takes two graph convolution layers and two SANEpools layers. To provide robust model performance, we perform five-fold cross-validation and report the accuracy averaged over five folds and the standard deviation of validation accuracies across the five folds.

4.3. Experimental Results

4.3.1. Predictive Performance

In the experiment, we demonstrate the effectiveness of SANEpools in predicting the synergy score of drug/drug combinations. Table 1 illustrates the experimental results. The experimental results indicate that SANEpools can accurately predict the synergy score of drug/drug combinations and achieve the state-of-the-art predictive performance among these competitive baselines.

Furthermore, (1) we find that SANEpools significantly improves the performance over flat GNNs (i.e., GIN, GCN, GAT, DAGNN), which indicates that hierarchical graph representation learning technique in molecular networks can provide informative graph

embedding with biologically meaning. (2) SANEpool outperforms other hierarchical graph pooling algorithms, and this observation indicates that incorporating edge information in the graph pooling is a potential future direction in the molecular network analysis, where graphs always have thousands of high-centrality nodes.

Table 1. Performance evaluation. Best results are in bold.

Model	NCI-DCD		GDSC-SDD		O'Neil-DCD	
	Pearson'r ↑	MSE ↓	Pearson'r ↑	RMSE ↓	Pearson'r ↑	RMSE ↓
DeepSynergy	0.589 ± 0.022	47.742 ± 2.950	0.703 ± 0.014	0.0166 ± 0.0020	0.537 ± 0.021	187.56 ± 16.75
DeepSignalingSynergy	0.631 ± 0.019	45.218 ± 1.889	0.744 ± 0.011	0.0143 ± 0.0012	0.598 ± 0.022	166.15 ± 19.56
TransSynergy	0.644 ± 0.023	46.219 ± 3.208	0.794 ± 0.022	0.0129 ± 0.0031	0.615 ± 0.020	160.19 ± 17.33
GIN	0.565 ± 0.042	51.732 ± 5.636	0.716 ± 0.015	0.0155 ± 0.0017	0.550 ± 0.019	184.58 ± 17.06
GCN	0.494 ± 0.049	58.585 ± 5.618	0.707 ± 0.014	0.0169 ± 0.0014	0.540 ± 0.024	187.84 ± 18.39
DAGNN	0.509 ± 0.025	57.827 ± 3.174	0.638 ± 0.016	0.0198 ± 0.0018	0.431 ± 0.023	213.28 ± 20.19
GAT	0.571 ± 0.031	50.995 ± 3.021	0.623 ± 0.013	0.0230 ± 0.0015	0.522 ± 0.017	189.27 ± 18.51
SAGpool	0.537 ± 0.031	53.125 ± 4.116	0.568 ± 0.011	0.0270 ± 0.0024	0.478 ± 0.016	197.69 ± 22.94
Diffpool	0.577 ± 0.022	52.449 ± 3.155	0.658 ± 0.014	0.0186 ± 0.0039	0.517 ± 0.026	191.27 ± 18.31
SANEpool (our model)	0.656 ± 0.016	44.352 ± 2.241	0.825 ± 0.009	0.0113 ± 0.0013	0.614 ± 0.019	159.29 ± 23.01

4.3.2. Interpretability

The interpretability of deep learning models has been a major limiting factor for the use of these models in real-world drug-combination synergy analysis since most usage cases require explanations of the features used in the model. There is a natural trade-off between the interpretability and the accuracy of decision models in application, and hence it is critical to find the balance between them. Currently, there are multiple deep learning models which take massive drug chemical structure information and predict the synergy score in a fully untransparent manner, such as DeepSynergy and MatchMarker. Although they can achieve expressive predictive performance, the lack of interpretability somehow limits the power of these models in real-world applications. Among the selected baselines, DeepSignalingSynergy is constructed based on the standard multiple layer perceptron model, and hence it is inherently not interpretable. Similarly, GCN and GIN follow the basic neighborhood aggregation framework that aggregates information from the neighborhood of each node and then updates the node feature of the node based on the aggregated feature and node feature itself, while such aggregation processes are not interpretable either. In contrast, attention-based deep learning models, such as GAT and TransSynergy, can provide interpretable conclusions. For instance, GAT provides interpretability by measuring the connection strength between genes and graphs in the input biomolecular graphs through the attention mechanism, and we can analyze the effect of drug nodes based the connection strength. Analogous to these interpretable models, in the next section, we will show that the interpretability of the proposed SANEpool model comes from its ability of computing the ‘synergic importance’ of each gene. The ‘synergic importance’ is computed as the expectation of its effect on the synergic score of drug combinations. Specifically, the effect can be measured by whether it is used in the synergic score prediction process (0/1 variable, i.e., whether the gene is selected by SANEpool model) or we can multiply the 0/1 variable with the synergy score. In this paper, we use the former definition. Hence, the ‘synergic importance’ can be used to detect genes with closer correlations to synergic drug combinations in each cell line by selecting genes whose ‘synergic importance’ is larger than a given threshold.

4.4. Statistical Analysis and Visualizations

In the experiment, we implement statistical analysis and visual analytics to reveal the interpretability of the SANEpool model. Here we use NCI-DCD as an example. Previous works [20,40] have emphasized that drug synergy is highly cell-line-specific/context-specific. Hence, we perform the cell-line-specific analysis. For each cancer cell line, there are multiple drug combinations targeting the cell line, some of which are synergic, while

others are not synergic. For each drug combination, SANepool can select top k genes for the prediction. Hence, we define the synergic/non-synergic importance score of a gene as the proportion of the gene is used by the SANepool model in the prediction when input drug pairs are synergistic/non-synergistic. For instance, if a gene node is never detected by any synergistic drug combination that targets on a specific cancer cell line, then its synergic importance score is 0 and is never used to predict the synergy score of synergic combinations. Figure 3 compares these scores computed by SANepool, and it illustrates that the patterns of synergic importance scores and non-synergic importance scores across genes are different in each cell line.

Next, we should decide whether the detected gene sub-networks for synergic drug combinations are significantly different from those of non-synergistic drug combinations. In order to do so, we can compare the distribution of the synergic importance scores on genes and the distribution of the non-synergic importance scores. If these two distributions are significantly different, we can infer that the detected gene sub-networks are also significantly different. Hence, we implement the Kolmogorov–Smirnov test (K-S test) to compare these distributions. In the K-S test, the null hypothesis assumes that two distributions are the same, and it computes a D statistic as well as a p -value corresponding to the D statistic. Then, we reject the null hypothesis if the p -value is less than the significance level (0.05). We provide details of the K-S test and relevant statistics in Appendix C. We implement the cell-line-based test. The cell-line based K-S test results are provided in Table 2, and it shows that the K-S test is significant for each cell line. Consequently, we use the difference in the synergic importance score and non-synergic importance score of each gene to determine whether the gene is selected in the core gene sub-network. That is, the gene is obtained in the core gene sub-network if the computed value is above a given threshold, like 0.1.

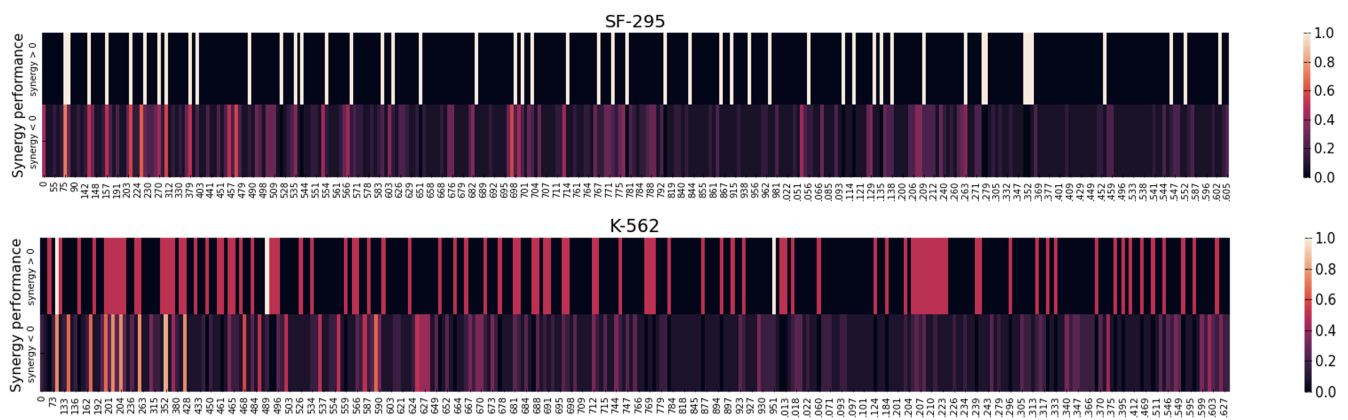


Figure 3. Synergic/non-synergic importance scores of all genes in cell line SF-295 and cell line K-562. X-axis is the gene index. The same genes in different cell lines share the same index.

Table 2. Cell -line-based K-S test results.

Cell Line	<i>p</i> Value	Cell Line	<i>p</i> Value	Cell Line	<i>p</i> Value	Cell Line	<i>p</i> Value	Cell Line	<i>p</i> Value
UACC-62	<0.001	NCI-H522	<0.001	HT29	0.002	MDA-MB-435	<0.001	A549 / ATCC	0.003
OVCAR-8	<0.001	HOP-62	0.003	HCT-15	<0.001	RPMI-8226	<0.001	MDA-MB-231 / ATCC	<0.001
OVCAR-3	<0.001	HS 578T	0.003	UO-31	<0.001	BT-549	0.005	UACC-257	<0.001
LOX IMVI	0.003	SW-620	<0.001	MCF7	<0.001	NCI-H460	<0.001	EKVX	<0.001
HOP-92	<0.001	SF-268	<0.001	K-562	0.007	T-47D	0.002	MDA-MB-468	<0.001
MALME-3M	<0.001	SK-MEL-5	<0.001	SF-295	0.004	NCI-H23	<0.001	OVCAR-4	0.002
SF-539	<0.001	U251	<0.001	PC-3	0.005	CAKI-1	0.007	HCT-116	<0.001
IGROV1	<0.001	SK-OV-3	0.006	A498	<0.001	NCI-H322M	<0.001	ACHN	<0.001
HL-60(TB)	0.005	KM12	<0.001	NCI-H226	<0.001	SK-MEL-28	<0.001	DU-145	0.004

In addition to the statistical analysis, we also use heat maps to show the difference between detected sub-networks of synergic drug combinations and non-synergic drug combinations. Here, we provide examples on cell line SF-295 and cell line K-562 in Figure 3,

and more examples are provided in Appendix A. In the heatmap, the values (synergy > 0 , synergy < 0) assigned to each gene are the cell-line-based synergic importance and non-synergic importance scores, which indicate the proportion that the gene is included in the gene sub-network (detected by SANEpool) of synergic drug combinations (non-synergic drug combinations) targeting the cell line. The synergy value of a gene measures the possibility that the information of the gene, such as the gene expression and gene copy number, contributes to the prediction process of the drug synergy score. Hence, the difference between cell-line-based synergy value (estimated probability that the gene is involved in the detected sub-network of drug combinations with synergy > 0) and cell-line-based non-synergy value (estimated probability that the gene is involved in the detected sub-network of drug combinations with synergy < 0) can reflect the synergic performance of the gene. That is, a larger difference indicates the gene contributes more to the synergic drug combination than non-synergic drug combinations. For instance, for cell line SK-562, the top 10 detected genes are SIN3A, ETS2, WNT10B, SLC8A1, MTOR, KLF2, RGS2, SESN3, NRG1, TNFRSF11A. Furthermore, these heatmaps also show that the difference between the detected genes of synergic drug combinations and of non-synergic drug combinations are significant.

Furthermore, we also visualize the interactions of drugs and genes in the detected core gene sub-network in Figures 4 and 5 to show that synergic drug combinations are more likely to target on the detected core gene sub-networks in each cell line. Figure 4 focuses on specific cell lines, while Figure 5 combines all cell lines and drug combinations. For each cell line, we plot all genes (i.e., red nodes) in the sub-networks (detected by SANEpool) of synergic drug combinations and then randomly sample synergic drug combinations (i.e., purple nodes) and some non-synergic drug combinations (i.e., blue nodes). Figure 4 illustrates that drugs in the non-synergic combinations are very unlikely to target on the core gene sub-network in each example cell line (e.g., SF-295, K-562). Figure 5 indicates that this observation can be extended to other cell lines.

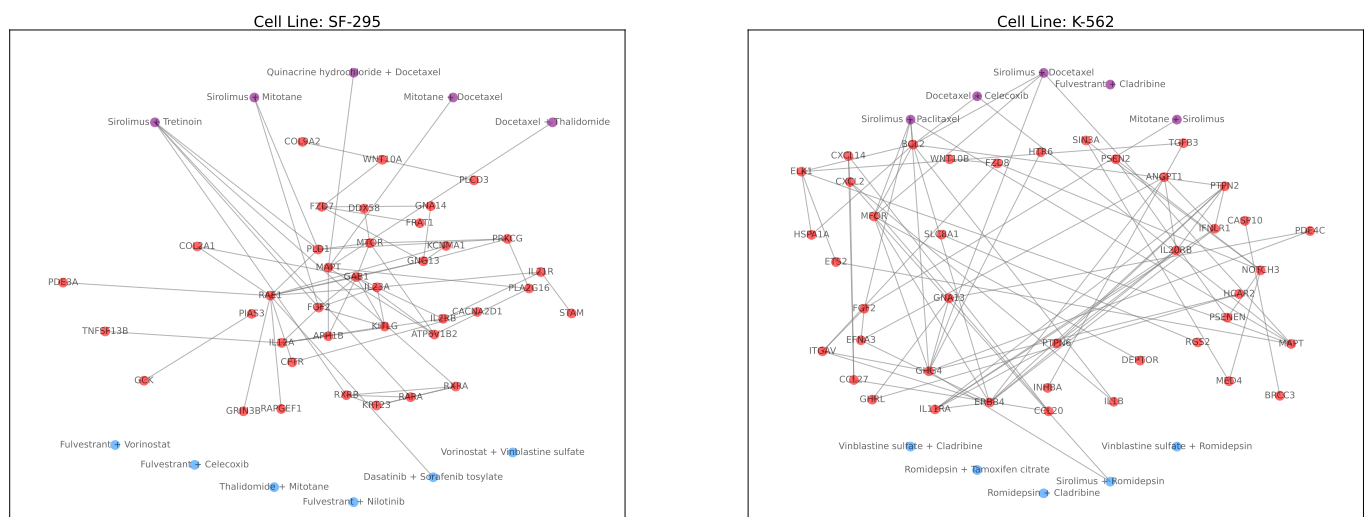


Figure 4. Visualization of drug–gene interactions on selected cell lines. In these graphs, red nodes represent genes in the subnetwork of all synergic drug combinations on the cell line, purple nodes are synergic drug pairs, blue nodes are randomly non-synergic drug pairs.

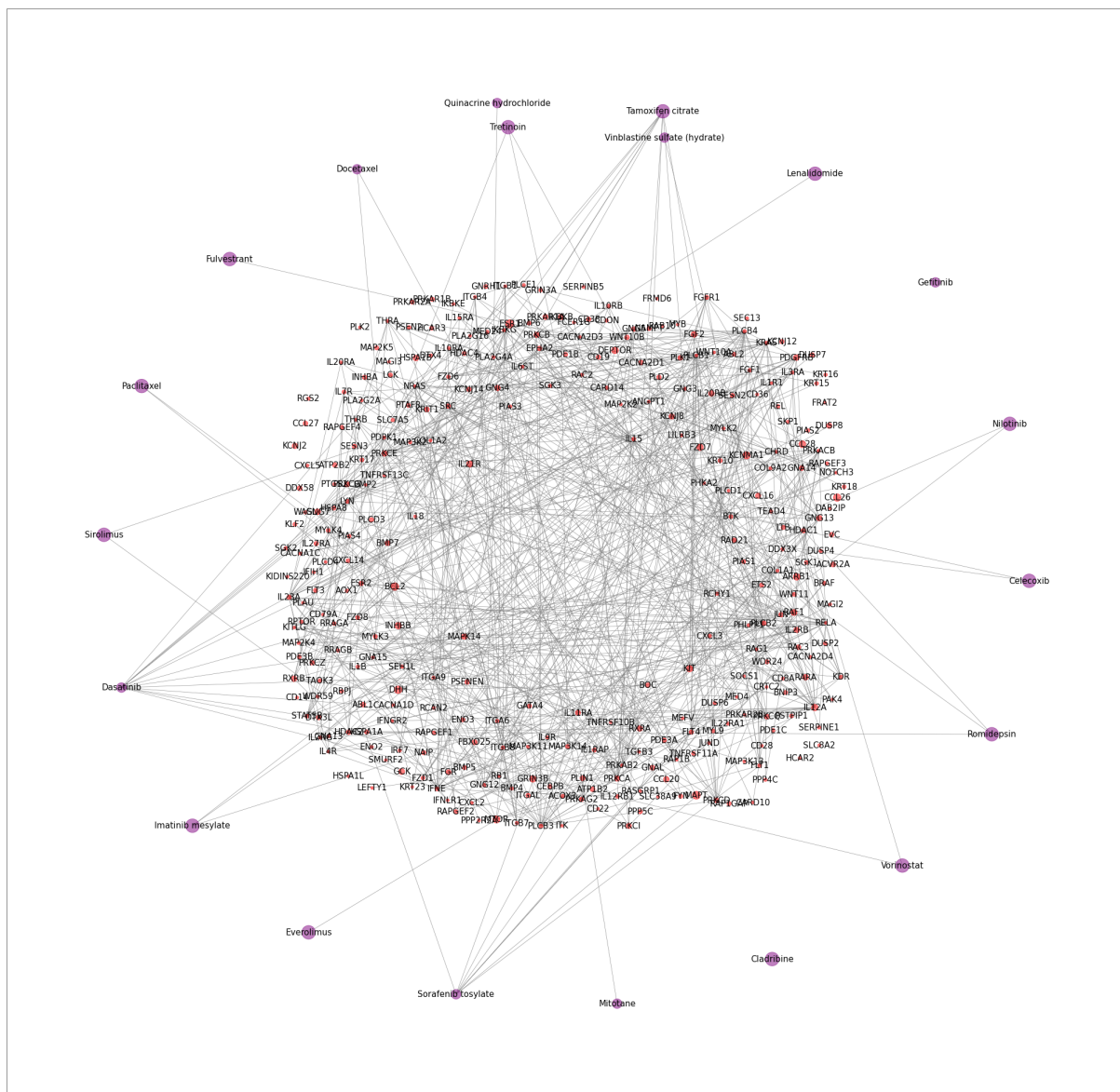


Figure 5. The figure describes the interactions of synergistic drugs combinations and genes in the detected core gene sub-network. Synergistic drug nodes are visualized as purple nodes, while red nodes represent genes in the detected core gene sub-network.

5. Conclusions

In this paper, we have proposed an interpretable GNN architecture called SANepool (self-attention-based node and edge pool) to predict the synergy score of drug combinations and to investigate the underlying mechanism of the synergy (MoS) by detecting salient molecular sub-networks. For each cell line and each drug combination, SANepool can detect a specific sub-network, and SANepool evaluates the contribution of each gene to synergistic drug combinations based on all detected (cell-line-based) sub-networks. Hence, cell-line-specific essential signaling gene targets are identified by SANepool. Furthermore, our observation also indicates that most synergistic drug combinations inhibit the core signaling network detected by SANepool. The current work is limited by the number of drug combinations and cell lines. In future work, more drug combination datasets with multi-omic data will be integrated to uncover the mechanism of the synergy of effective drug combinations.

Author Contributions: Conceptualization, Z.D., Y.C. and F.L.; methodology, Z.D. and F.L.; software, Z.D.; validation, Z.D.; data curation, H.Z., P.R.O.P. and F.L.; writing—original draft preparation, Z.D.; writing—review and editing, Z.D., H.Z., Y.C. and F.L.; visualization, Z.D.; supervision, Y.C., P.R.O.P. and F.L.; project administration, Y.C., P.R.O.P. and F.L.; funding acquisition, Y.C., P.R.O.P. and F.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work is partially supported by the Children’s Discovery Institute (CDI) M-II-2019-802, and United States National Library of Medicine (NLM) 1R01LM013902-01A1 to Fuhai Li.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Our source codes, which include datasets and implementation pipelines for reproducibility, are available at <https://github.com/zehao-dong> (accessed on 17 July 2023).

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Cell-Line Based Visualization Results

In this section, we provide the assigned synergy value (estimated probability that the gene is involved in the detected sub-network of drug combinations with synergy > 0) and assigned non-synergy value (estimated probability that the gene is involved in the detected sub-network of drug combinations with synergy < 0) of all genes for each cell line. As Figure A1 illustrates, the synergy value of a gene, which measures its importance in the prediction of synergy score, shows a significant difference in each cell line between synergistic drug combinations (synergy > 0) and non-synergistic drug combinations (synergy < 0).

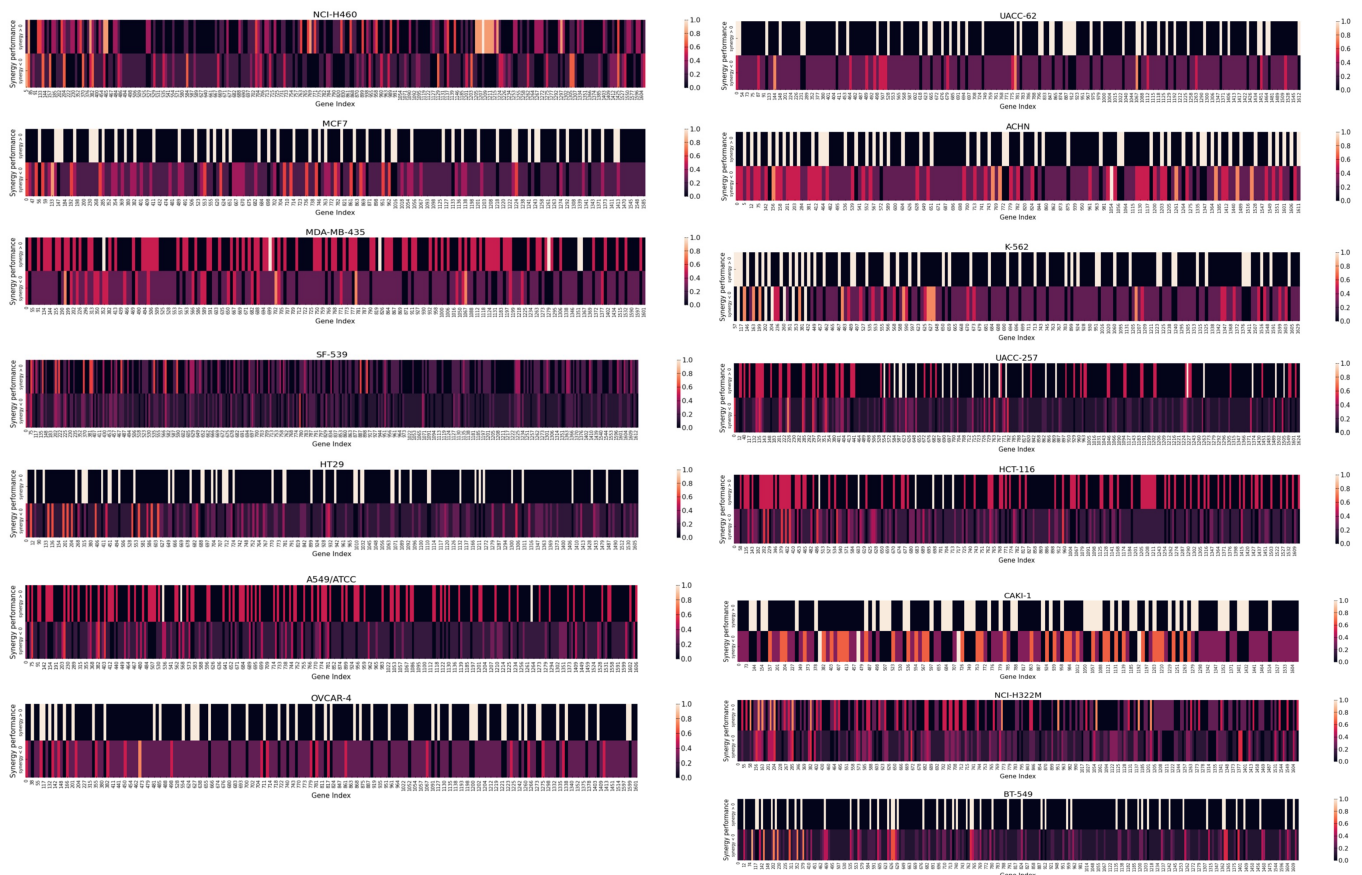


Figure A1. Comparison of gene synergy values in all cell-lines.

Appendix B. Cell Lines and FDA Approved Drugs

This section introduces the cell lines and drugs used in the dataset NCI-DCD. For dataset O'Neil-DCD and GDSC-SDD, the relevant information is available in previous works [5,40].

Signaling pathways used to formulate the input graphs: MAPK, FoxO, TGF-beta, T-cell receptor, Adipocytokine, ErbB, Sphingolipid, VEGF, B-cell receptor, Oxytocin, Ras, Phospholipase D, Apelin, Fc epsilon RI, Glucagon, Rap1, p53, Hippo, TNF, Relaxin, Calcium, mTOR, Toll-like receptor, Neurotrophin, AGE-RAGE, cGMP-PKG, PI3K-Akt, NOD-like receptor, Insulin, Cell cycle, cAMP, AMPK, RIG-I-like receptor, GnRH, Chemokine, Wnt, C-type lectin receptor, Estrogen, NF-kappa B, Notch, JAK-STAT, Prolactin, HIF-1, Hedgehog, IL-17, thyroid hormone.

FDA approved drugs used to formulate the input graphs: Celecoxib, Gefitinib, Quinacrine hydrochloride, Tretinoin, Cladribine, Imatinib mesylate, Romidepsin, Vinblastine sulfate (hydrate), Dasatinib, Lenalidomide, Sirolimus, Vorinostat, Docetaxel, Mitotane, Sorafenib tosylate, Thalidomide, Everolimus, Nilotinib, Tamoxifen Citrate, Paclitaxel, Fulvestrant.

Appendix C. Details of K-S Test

Here, we introduce more details of the cell-line-based K-S test. The objective is to test whether the detected sub-gene network (by SANEpools model) is significantly different for synergic drug combinations (synergy score > 0) and non-synergic drug combinations (synergy score < 0). It can be difficult to directly test the hypothesis. However, since SANEpools sorts the attention score, Att , of genes to select gene in the 'core sub-gene network', each gene has different probability to be selected by synergic and non-synergic drug combinations. Hence, we propose to compare the distribution of selected genes for synergic and non-synergic drug combinations in each cell line. Next, we need some observations to perform the K-S test. For each cell line, suppose that there are N synergic and M non-synergic drug combinations on the cell line. Then, for synergic drug combinations, each observation will randomly sample $0.8 \times N$ synergic samples, and then use the proportion of each gene included in the detected sub-gene network (by SANEpools) as its probability of being selected. Then we get an observation ps_i , which is a vector of probability of being selected by synergic drug combinations for all genes. Similarly, we can obtain an observation pns_i from $0.8 \times M$ non-synergic samples. Then, we sample 10 observations for both synergic and non-synergic drug combinations: ps_i, pns_i for $i = 1, 2, \dots, 10$. After that, we simply need to combine these observations and compute the empirical cumulative distribution function (cdf) F . Then, we can perform the two-sample K-S test (i.e., $D_n = \max(|F_{exp} - F_{obs}|)$), where F_{exp} and F_{obs} are the empirical cumulative distribution functions computed based on ps_i and pns_i .

Appendix D. Advantage of Using Pre-Defined Order of Genes in SANEpools

For general graph learning problems, it can be difficult to find reasonable pre-defined indexes for nodes in graphs, as the problem is equivalent to the graph isomorphism problem, which is known to be NP-hard. However, for gene networks, since each gene at most appears once in each network, we can universally assign each possible gene (that appears in at least one network/graph in the data set) a unique (pre-defined) index. For instance, we can collect all possible genes and lexicographically sort their names, and then the order of genes in the sorting operation can be used as the index. Consequently, the pre-defined index is equivalent to the gene. It does not matter if we change the pre-defined index system once these indexes can injectively distinguish genes and are consistent among gene networks. The main advantage is to reduce the space complexity. Since the gene networks usually contain lots of genes (i.e., a very large n), then, to store all possible edge pairs in the layer t , we require a tensor T with the shape of $R^{n \times n \times 2 \times h_t}$, where h_t is the size of node representation vectors in layer t and usually is selected from [32, 64, 128, 256]. And then, an MLP is used to learn the edge attention matrix from T . In contrast, using

pre-defined indexes can reduce the size of T to $R^{n \times n}$, and no MLP is needed in this step. Hence, we can significantly shrink the model/algorithm complexity.

Appendix E. Details of the Overall Model Architecture

The overall model architecture is composed of SANEpool layer → graph convolution layer → batch normalization layer → SANEpool layer → graph convolution layer → batch normalization layer → readout layer (Equation (6)). Following this flow, the proposed model will output a predicted synergy score value for each input network/graph. In addition to the SANEpool layers and graph convolution layers, batch normalization layers are used to regularize the model training and to avoid overfitting.

References

- Hopkins, A.L. Network pharmacology: The next paradigm in drug discovery. *Nat. Chem. Biol.* **2008**, *4*, 682–690. [[CrossRef](#)]
- Podolsky, S.H.; Greene, J.A. Combination drugs—Hype, harm, and hope. *N. Engl. J. Med.* **2011**, *365*, 488–491. [[CrossRef](#)]
- Chandrasekaran, S.; Cokol-Cakmak, M.; Sahin, N.; Yilancioglu, K.; Kazan, H.; Collins, J.J.; Cokol, M. Chemogenomics and orthology-based design of antibiotic combination therapies. *Mol. Syst. Biol.* **2016**, *12*, 872. [[CrossRef](#)]
- Radic-Sarikas, B.; Tsaou, K.P.; Emdal, K.B.; Papamarkou, T.; Huber, K.V.; Mutz, C.; Toretzky, J.A.; Bennett, K.L.; Olsen, J.V.; Brunak, S.; et al. Combinatorial drug screening identifies Ewing sarcoma—Specific sensitivities. *Mol. Cancer Ther.* **2017**, *16*, 88–101. [[CrossRef](#)]
- O’Neil, J.; Benita, Y.; Feldman, I.; Chenard, M.; Roberts, B.; Liu, Y.; Li, J.; Kral, A.; Lejnine, S.; Loboda, A.; et al. An unbiased oncology compound screen to identify novel combination strategies. *Mol. Cancer Ther.* **2016**, *15*, 1155–1162. [[CrossRef](#)]
- Devita, V.T., Jr.; Young, R.C.; Canellos, G.P. Combination versus single agent chemotherapy: A review of the basis for selection of drug treatment of cancer. *Cancer* **1975**, *35*, 98–110. [[CrossRef](#)]
- Crino, L.; Scagliotti, G.; Marangolo, M.; Figoli, F.; Clerici, M.; De Marinis, F.; Salvati, F.; Cruciani, G.; Dogliotti, L.; Pucci, F.; et al. Cisplatin-gemcitabine combination in advanced non-small-cell lung cancer: A phase II study. *J. Clin. Oncol.* **1997**, *15*, 297–303. [[CrossRef](#)]
- Carew, J.S.; Giles, F.J.; Nawrocki, S.T. Histone deacetylase inhibitors: Mechanisms of cell death and promise in combination cancer therapy. *Cancer Lett.* **2008**, *269*, 7–17. [[CrossRef](#)]
- Shuhendler, A.J.; Cheung, R.Y.; Manias, J.; Connor, A.; Rauth, A.M.; Wu, X.Y. A novel doxorubicin-mitomycin C co-encapsulated nanoparticle formulation exhibits anti-cancer synergy in multidrug resistant human breast cancer cells. *Breast Cancer Res. Treat.* **2010**, *119*, 255–269. [[CrossRef](#)]
- Mott, B.T.; Eastman, R.T.; Guha, R.; Sherlach, K.S.; Siriwardana, A.; Shinn, P.; McKnight, C.; Michael, S.; Lacerda-Queiroz, N.; Patel, P.R.; et al. High-throughput matrix screening identifies synergistic and antagonistic antimalarial drug combinations. *Sci. Rep.* **2015**, *5*, 13891. [[CrossRef](#)]
- Griner, L.A.M.; Guha, R.; Shinn, P.; Young, R.M.; Keller, J.M.; Liu, D.; Goldlust, I.S.; Yasgar, A.; McKnight, C.; Boxer, M.B.; et al. High-throughput combinatorial screening identifies drugs that cooperate with ibrutinib to kill activated B-cell-like diffuse large B-cell lymphoma cells. *Proc. Natl. Acad. Sci. USA* **2014**, *111*, 2349–2354. [[CrossRef](#)] [[PubMed](#)]
- Holbeck, S.L.; Camalier, R.; Crowell, J.A.; Govindharajulu, J.P.; Hollingshead, M.; Anderson, L.W.; Polley, E.; Rubinstein, L.; Srivastava, A.; Wilsker, D.; et al. The National Cancer Institute ALMANAC: A comprehensive screening resource for the detection of anticancer drug pairs with enhanced therapeutic activity. *Cancer Res.* **2017**, *77*, 3564–3576. [[CrossRef](#)] [[PubMed](#)]
- Yang, J.C.H.; Mok, T.; Han, B.; Orlando, M.; Puri, T.; Park, K. A review of regimens combining pemetrexed with an epidermal growth factor receptor tyrosine kinase inhibitor in the treatment of advanced nonsquamous non-small-cell lung cancer. *Clin. Lung Cancer* **2018**, *19*, 27–34. [[CrossRef](#)] [[PubMed](#)]
- Zhang, T.; Zhang, L.; Payne, P.R.; Li, F. Synergistic drug combination prediction by integrating multiomics data in deep learning models. In *Translational Bioinformatics for Therapeutic Development*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 223–238.
- Janizek, J.D.; Celik, S.; Lee, S.I. Explainable machine learning prediction of synergistic drug combinations for precision cancer medicine. *bioRxiv* **2018**, 331769. [[CrossRef](#)]
- Chen, T.; Guestrin, C. Xgboost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 785–794.
- Preuer, K.; Lewis, R.P.; Hochreiter, S.; Bender, A.; Bulusu, K.C.; Klambauer, G. DeepSynergy: Predicting anti-cancer drug synergy with Deep Learning. *Bioinformatics* **2018**, *34*, 1538–1546. [[CrossRef](#)] [[PubMed](#)]
- Sidorov, P.; Naulaerts, S.; Ariey-Bonnet, J.; Pasquier, E.; Ballester, P.J. Predicting synergism of cancer drug combinations using NCI-ALMANAC data. *Front. Chem.* **2019**, *7*, 509. [[CrossRef](#)]
- Kuru, H.I.; Tastan, O.; Cicek, E. MatchMaker: A deep learning framework for drug synergy prediction. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2021**, *19*, 2334–2344. [[CrossRef](#)]
- Hosseini, S.R.; Zhou, X. CCSynergy: An integrative deep-learning framework enabling context-aware prediction of anti-cancer drug synergy. *Briefings Bioinform.* **2023**, *24*, bbac588. [[CrossRef](#)]

21. Liu, Q.; Xie, L. TranSynergy: Mechanism-driven interpretable deep neural network for the synergistic prediction and pathway deconvolution of drug combinations. *PLoS Comput. Biol.* **2021**, *17*, e1008653. [[CrossRef](#)]
22. Zhang, P.; Tu, S.; Zhang, W.; Xu, L. Predicting cell line-specific synergistic drug combinations through a relational graph convolutional network with attention mechanism. *Briefings Bioinform.* **2022**, *23*, bbac403. [[CrossRef](#)]
23. Wang, J.; Liu, X.; Shen, S.; Deng, L.; Liu, H. DeepDDS: Deep graph neural network with attention mechanism to predict synergistic drug combinations. *Briefings Bioinform.* **2022**, *23*, bbab390. [[CrossRef](#)] [[PubMed](#)]
24. Carvalho, D.V.; Pereira, E.M.; Cardoso, J.S. Machine learning interpretability: A survey on methods and metrics. *Electronics* **2019**, *8*, 832. [[CrossRef](#)]
25. Burkart, N.; Huber, M.F. A survey on the explainability of supervised machine learning. *J. Artif. Intell. Res.* **2021**, *70*, 245–317. [[CrossRef](#)]
26. Monti, F.; Bronstein, M.M.; Bresson, X. Geometric matrix completion with recurrent multi-graph neural networks. *arXiv* **2017**, arXiv:1704.06803.
27. Ying, R.; He, R.; Chen, K.; Eksombatchai, P.; Hamilton, W.L.; Leskovec, J. Graph convolutional neural networks for web-scale recommender systems. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, London, UK, 19–23 August 2018; pp. 974–983.
28. Fout, A.M. Protein iNterface Prediction Using Graph Convolutional Networks. Ph.D. Thesis, Colorado State University, Fort Collins, CO, USA, 2017.
29. Zitnik, M.; Agrawal, M.; Leskovec, J. Modeling polypharmacy side effects with graph convolutional networks. *Bioinformatics* **2018**, *34*, i457–i466. [[CrossRef](#)]
30. Dong, Z.; Zhang, M.; Li, F.; Chen, Y. Pace: A parallelizable computation encoder for directed acyclic graphs. In Proceedings of the International Conference on Machine Learning, Baltimore, MD, USA, 17–23 July 2022; PMLR: London, UK, 2022; pp. 5360–5377.
31. Dong, Z.; Cao, W.; Zhang, M.; Tao, D.; Chen, Y.; Zhang, X. CktGNN: Circuit Graph Neural Network for Electronic Design Automation. In Proceedings of the Eleventh International Conference on Learning Representations, Kigali, Rwanda, 1–5 May 2023.
32. Hamilton, W.L.; Ying, R.; Leskovec, J. Inductive representation learning on large graphs. *arXiv* **2017**, arXiv:1706.02216.
33. Schütt, K.T.; Kindermans, P.J.; Sauceda, H.E.; Chmiela, S.; Tkatchenko, A.; Müller, K.R. SchNet: A continuous-filter convolutional neural network for modeling quantum interactions. *arXiv* **2017**, arXiv:1706.08566.
34. Zhang, M.; Chen, Y. Link prediction based on graph neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 3–8 December 2018; Volume 31, pp. 5165–5175.
35. Dai, H.; Dai, B.; Song, L. Discriminative embeddings of latent variable models for structured data. In Proceedings of the International Conference on Machine Learning, New York, NY, USA, 20–22 June 2016; PMLR: London, UK, 2016; pp. 2702–2711.
36. Zhang, M.; Cui, Z.; Neumann, M.; Chen, Y. An end-to-end deep learning architecture for graph classification. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018.
37. Ying, Z.; You, J.; Morris, C.; Ren, X.; Hamilton, W.; Leskovec, J. Hierarchical graph representation learning with differentiable pooling. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 3–8 December 2018; Volume 31.
38. Sanchez-Vega, F.; Mina, M.; Armenia, J.; Chatila, W.K.; Luna, A.; La, K.C.; Dimitriadou, S.; Liu, D.L.; Kantheti, H.S.; Saghafinia, S.; et al. Oncogenic signaling pathways in the cancer genome atlas. *Cell* **2018**, *173*, 321–337. [[CrossRef](#)]
39. Pan, R.; Ruvolo, V.; Mu, H.; Levenson, J.D.; Nichols, G.; Reed, J.C.; Konopleva, M.; Andreeff, M. Synthetic lethality of combined Bcl-2 inhibition and p53 activation in AML: Mechanisms and superior antileukemic efficacy. *Cancer Cell* **2017**, *32*, 748–760. [[CrossRef](#)]
40. Jaaks, P.; Coker, E.A.; Vis, D.J.; Edwards, O.; Carpenter, E.F.; Leto, S.M.; Dwane, L.; Sassi, F.; Lightfoot, H.; Barthorpe, S.; et al. Effective drug combinations in breast, colon and pancreatic cancer cells. *Nature* **2022**, *603*, 166–173. [[CrossRef](#)]
41. Schlichtkrull, M.; Kipf, T.N.; Bloem, P.; Berg, R.v.d.; Titov, I.; Welling, M. Modeling relational data with graph convolutional networks. In Proceedings of the European Semantic Web Conference, Monterey, CA, USA, 8–12 October 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 593–607.
42. Verma, S.; Zhang, Z.L. Graph capsule convolutional neural networks. *arXiv* **2018**, arXiv:1805.08090.
43. Niepert, M.; Ahmed, M.; Kutzkov, K. Learning convolutional neural networks for graphs. In Proceedings of the International Conference on Machine Learning, New York, NY, USA, 20–22 June 2016; PMLR: London, UK, 2016; pp. 2014–2023.
44. Yao, J.; Arcila, M.E.; Ladanyi, M.; Hechtman, J.F. Pan-Cancer Biomarkers: Changing the Landscape of Molecular Testing. *Arch. Pathol. Lab. Med.* **2021**, *145*, 692–698. [[CrossRef](#)] [[PubMed](#)]
45. Jin, G.; Zhao, H.; Zhou, X.; Wong, S.T. An enhanced Petri-net model to predict synergistic effects of pairwise drug combinations from gene microarray data. *Bioinformatics* **2011**, *27*, i310–i316. [[CrossRef](#)] [[PubMed](#)]
46. Wu, Z.; Zhao, X.M.; Chen, L. A systems biology approach to identify effective cocktail drugs. *BMC Syst. Biol.* **2010**, *4*, S7. [[CrossRef](#)]
47. Chen, D.; Zhang, H.; Lu, P.; Liu, X.; Cao, H. Synergy evaluation by a pathway–pathway interaction network: A new way to predict drug combination. *Mol. BioSyst.* **2016**, *12*, 614–623. [[CrossRef](#)]
48. Li, P.; Huang, C.; Fu, Y.; Wang, J.; Wu, Z.; Ru, J.; Zheng, C.; Guo, Z.; Chen, X.; Zhou, W.; et al. Large-scale exploration and analysis of drug combinations. *Bioinformatics* **2015**, *31*, 2007–2016. [[CrossRef](#)]

49. Xu, K.J.; Hu, F.Y.; Song, J.; Zhao, X.M. Exploring drug combinations in a drug-cocktail network. In Proceedings of the 2011 IEEE International Conference on Systems Biology (ISB), Zhuhai, China, 2–4 September 2011; IEEE: Piscataway, NJ, USA, 2011; pp. 382–387.
50. Yin, N.; Ma, W.; Pei, J.; Ouyang, Q.; Tang, C.; Lai, L. Synergistic and antagonistic drug combinations depend on network topology. *PLoS ONE* **2014**, *9*, e93960. [[CrossRef](#)]
51. Dong, Z.; Zhang, H.; Chen, Y.; Li, F. Interpretable Drug Synergy Prediction with Graph Neural Networks for Human-AI Collaboration in Healthcare. *arXiv* **2021**, arXiv:2105.07082.
52. Parikh, A.P.; Täckström, O.; Das, D.; Uszkoreit, J. A decomposable attention model for natural language inference. *arXiv* **2016**, arXiv:1606.01933.
53. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
54. Yan, S.; Zheng, Y.; Ao, W.; Zeng, X.; Zhang, M. Does unsupervised architecture representation learning help neural architecture search? *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 12486–12498.
55. Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; Bengio, Y. Graph Attention Networks. *arXiv* **2018**, arXiv:1710.10903.
56. Cheng, J.; Dong, L.; Lapata, M. Long short-term memory-networks for machine reading. *arXiv* **2016**, arXiv:1601.06733.
57. Zhang, H.; Goodfellow, I.; Metaxas, D.; Odena, A. Self-attention generative adversarial networks. In Proceedings of the International Conference on Machine Learning, Long Beach, CA, USA, 9–15 June 2019; PMLR: London, UK, 2019; pp. 7354–7363.
58. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 6000–6010.
59. Lee, J.; Lee, I.; Kang, J. Self-attention graph pooling. In Proceedings of the International Conference on Machine Learning, Long Beach, CA, USA, 9–15 June 2019; PMLR: London, UK, 2019; pp. 3734–3743.
60. Defferrard, M.; Bresson, X.; Vandergheynst, P. Convolutional neural networks on graphs with fast localized spectral filtering. *arXiv* **2016**, arXiv:1606.09375.
61. Kipf, T.N.; Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv* **2016**, arXiv:1609.02907.
62. Cangea, C.; Veličković, P.; Jovanović, N.; Kipf, T.; Liò, P. Towards sparse hierarchical graph classifiers. *arXiv* **2018**, arXiv:1811.01287.
63. Hornik, K.; Stinchcombe, M.; White, H. Multilayer feedforward networks are universal approximators. *Neural Netw.* **1989**, *2*, 359–366. [[CrossRef](#)]
64. Hornik, K. Approximation capabilities of multilayer feedforward networks. *Neural Netw.* **1991**, *4*, 251–257. [[CrossRef](#)]
65. Zhang, M.; Chen, Y. Inductive matrix completion based on graph neural networks. *arXiv* **2019**, arXiv:1904.12058.
66. Bento, A.P.; Hersey, A.; Félix, E.; Landrum, G.; Gaulton, A.; Atkinson, F.; Bellis, L.J.; De Veij, M.; Leach, A.R. An open source chemical structure curation pipeline using RDKit. *J. Cheminform.* **2020**, *12*, 51. [[CrossRef](#)]
67. Wishart, D.S.; Feunang, Y.D.; Guo, A.C.; Lo, E.J.; Marcu, A.; Grant, J.R.; Sajed, T.; Johnson, D.; Li, C.; Sayeeda, Z.; et al. DrugBank 5.0: A major update to the DrugBank database for 2018. *Nucleic Acids Res.* **2018**, *46*, D1074–D1082. [[CrossRef](#)] [[PubMed](#)]
68. Kanehisa, M.; Goto, S. KEGG: Kyoto encyclopedia of genes and genomes. *Nucleic Acids Res.* **2000**, *28*, 27–30. [[CrossRef](#)] [[PubMed](#)]
69. Feng, J.; Zhang, H.; Li, F. Investigating the relevance of major signaling pathways in cancer survival using a biologically meaningful deep learning model. *BMC Bioinform.* **2021**, *22*, 47. [[CrossRef](#)] [[PubMed](#)]
70. Barretina, J.; Caponigro, G.; Stransky, N.; Venkatesan, K.; Margolin, A.A.; Kim, S.; Wilson, C.J.; Lehár, J.; Kryukov, G.V.; Sonkin, D.; et al. The Cancer Cell Line Encyclopedia enables predictive modelling of anticancer drug sensitivity. *Nature* **2012**, *483*, 603–607. [[CrossRef](#)] [[PubMed](#)]
71. Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; Bengio, Y. Graph Attention Networks. In Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018.
72. Xu, K.; Hu, W.; Leskovec, J.; Jegelka, S. How powerful are graph neural networks? *arXiv* **2018**, arXiv:1810.00826.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.