JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

DRUG TARGET INTERACTION PREDICTION USING GRAPH CONVOLUTIONAL NEURAL NETWORK

Kandukuri Sree Sai Sampat Gupta, Cherukuri Sai Manvija, Annepu Rohit, Telagani Siva Lakshman Ajith, Dr. D. Rama Krishna

Department of Computer Science Engineering, GITAM Deemed to be University, Visakhapatnam, India-530045

Abstract: DTI-GCNN (Drug-Target Interaction Prediction with Graph Convolution Neural Networks) presents an innovative framework for predicting drug-target interactions, which is crucial in proteomics and pharmaceutical research. Traditional methods often overlook the intricate graph structure of DTI networks, but DTI-GCNN efficiently utilizes this complexity. By treating DTI networks as graphs, the model captures both local and global interactions through graph convolutional layers, effectively integrating structural and semantic information. This approach showcases the potential of GCNNs to enhance DTI prediction accuracy, offering valuable insights for drug discovery and precision medicine.

Keywords: Graph Convolutional Neural Networks (GCNNs), Deep learning, Webscraping, Drug target interaction

I.INTRODUCTION

Drug-Target Interaction (DTI) prediction is the process of predicting how drugs will interact with the proteins that serve as their targets and is an important task in medicinal research and bioinformatics. Drugs are chemical substances that are used in biological systems to alter the activity of particular proteins, which are essential molecules engaged in a variety of cellular functions. When a medication attaches itself to a target protein, it can alter the protein's expression, activity, or function. This is known as the drug-target protein interaction.

The biological molecules known as proteins are the ones that drug compounds target. These proteins are necessary for several cellular functions, such as structural support, enzyme catalysis, and signal transmission. The project uses a graph network structure to describe proteins as nodes, where each node is associated with a particular protein target.

In this project, we seek to take advantage of the structural and functional parallels between drugs and proteins stored in the DTI network to predict new drug-target interactions precisely. Our model, called DTI-GCNN, learns to extract informative features from the graph representation by training a GCNN on a large dataset of known drug-target interactions. This allows it to effectively generalise to new drug-target pairings. A Graph Convolutional Neural Network (GCNN) is a type of neural network architecture developed to process graph-structured data. By extending the ideas of convolutional neural networks (CNNs) to graphs, deep learning techniques can be applied to applications involving graph-structured data, such the prediction of drug-target interactions (DTIs).

In conclusion, the goal of this project is to improve drug-protein interaction network prediction by utilising GCNNs' ability to learn from the intricate graph structure of drug-protein interaction networks. Our method has the potential to speed up drug discovery and development procedures by precisely forecasting novel drug-target interactions, which could ultimately result in the discovery of new treatments for a range of illnesses and ailments.

II.LITERATURE SURVEY

Title: EmbedDTI: Enhancing the Molecular Representations via Sequence Embedding and Graph Convolutional Network for the Prediction of Drug-Target Interaction (2021)
Publisher: Yuan Jin, Jiarui Lu, Runhan Shi, Yang Yang
This model improves how input target and compound information are represented. It represents every drug molecule as a graph of substructures in addition to a graph of atoms. It separates substructures and extracts their features using algorithms. It also uses word embedding techniques to pre-train amino acid sequences and a deep CNN to understand high-level abstract characteristics of proteins.

Title: Drug-Target Interaction Prediction with Graph Attention networks (2021)
Publisher: Haiyang Wang, Guangyu Zhou, Siqi Liu, Jyun-Yu Jiang, Wei Wang
This paper builds a drug-target heterogeneous network by proposing a drug-target interaction anticipation model. Based on similarities between medications or targets and recognized DTIs, it weights the edges and uses GAN's

Title: Predicting Drug−Target Interaction Using a Novel Graph Neural Network with 3D Structure-Embedded Graph Representation
Publisher: Jaechang Lim, Seongok Ryu, Kyubyong Park, Yo Joong Choe, Jiyeon Ham, Woo Youn Kim
Devises from the widely used CNN model and uses a graph convolution method. This method offers advantages in computation time and efficiency and is specifically intended for protein-ligand binding affinity prediction.

Title: Predicting Biomedical Interactions With Higher-Order Graph Convolutional Networks
Publisher: Kishan KC, Rui Li, Feng Cui, Anne R. Haake.
A deep graph convolutional network (HOGCN) created especially for biological contact prediction is presented in this paper. In order to learn and combine feature representations of neighbors at different sizes while taking higher-order neighborhood information into account, it employs a higher-order graph convolutional layer.

III.PROPOSED METHODOLOGY

1. **Data Collection:**

In the initial step of the system methodology, the data required for the project is to be collected from various sources.

Here are the key steps:

I. **Identifying Data Sources:** Examine relevant databases, repositories, and sources for drug-target interaction information. Determine the availability and accessibility of data from several sources.

II. **Data Validation and Curation:** Validate and clean the obtained data to ensure it is consistent, accurate, and comprehensive. Address missing values, duplicates, and inconsistencies in the extracted data. Curate the data to remove noise and useless information.

```
1    ,Drug_ID,Protein_ID,Label
2    0,DB00818,P18507,1
3    1,DB00421,Q08289,1
4    2,DB05930,DB03306,0
5    3,DB05407,DB05023,0
6    4,Q9BY49,P05771,0
7    5,DB03059,P30084,1
8    6,DB00159,Q8NER1,1
9    7,DB00711,P23219,1
10   8,DB00222,Q14654,1
11   9,DB00690,P78334,1
12   10,DB08250,DB07882,0
13   11,DB00421,O60840,1
14   12,DB02011,DB05227,0
15   13,DB02215,Q9UKQ2,1
16   14,DB02684,DB04818,0
17   15,DB07213,O14757,1
18   16,DB07574,P49759,0
19   17,DB00719,P35367,1
20   18,DB00818,P11509,1
```

Fig 3.1: This is representation of collected data for training, testing and validating.

This is the initial dataset which is in the CSV format with 3 attributes:

- Drug_ID

- Protein_ID
- Label

Initially, the data is having 30277 DTI.
The features of the dataset are (Protein ID, Drug ID and Label). But for GCN we need graph structures so we should preprocess the data which is shown in the next step.

2. Data Preprocessing:

In the second step of the system methodology, the DTI data is to be pre-processed and the ID's are then converted into Drug formulas or smiles and protein sequences.

This involves several key components:

I. **Cleaning and Formatting:** Clean up and preprocess the extracted data, which includes drug names, target protein identifiers, and interaction labels. Standardise data formats and representations to ensure consistency.

II. **Web scraping:** Develop web scraping scripts or tools that will retrieve drug SMILES (Simplified Molecular Input Line Entry System) formulas and protein sequences from the suggested data sources. In Python, use web scraping packages like BeautifulSoup or Scrapy to crawl through the chosen databases' web pages, identify important information, and extract it into structured data formats.

SMILES stands for "Simplified Molecular Input Line Entry System." It is a notation system that represents chemical structures in a clear and human-readable manner. SMILES strings express chemical structures using a linear sequence of characters, including letters, numbers, and symbols, based on the connectivity of atoms inside the molecule.

Protein sequences are the linear amino acid chains that make up a protein's main structure. Amino acids are the building blocks of proteins, which are joined together by peptide bonds to create polypeptide chains. Each protein has a distinct sequence of amino acids that define its shape, function, and interactions with other molecules in the cell.

III. **Data Mapping:** Create a mapping between medication IDs and their formulas, as well as protein IDs and their sequences. As you collect medication formulas and protein sequences from the web, keep a mapping table or dictionary that links each drug ID to its formula and each protein ID to its sequence.

IV. **Data Replacement:** Iterate through the existing dataset of drug and protein IDs.

For each drug ID, look up the appropriate formula in the mapping table and substitute the drug ID with the formula.

For each protein ID, look up the appropriate sequence in the mapping table and substitute the protein ID with the sequence.

V. **Data Integration:** Once the drug IDs and protein IDs have been replaced with drug formulas and protein sequences, the new data should be added to the analytic workflow. We must ensure that the revised dataset is compatible with the following processing steps: feature engineering, model training, and evaluation.

VI. **Quality Assurance:** Perform quality assurance checks to ensure that the revised dataset is accurate and full. Check that the substituted medication formulae and protein sequences correspond to the expected values obtained throughout the web scraping procedure.

3. Feature Engineering:

In the third step of the system methodology, Feature engineering is performed. Feature engineering or extraction plays a critical role in turning raw data, such as drug formulas and protein sequences, into a format suited for graph-based analysis, especially for tasks such as drug-target interaction (DTI) prediction. In the context of graph convolutional neural networks (GCNNs) for DTI prediction, feature engineering includes encoding medicines and proteins as nodes in a graph, with their associated features representing the node attributes. Here are the key steps in the feature engineering:

I.Drug Representation:

a. **Molecular Fingerprints:** Molecular fingerprints record medications' structural traits as binary or integer vectors, indicating the presence or absence of specific substructures or chemical properties. Extended Connectivity Fingerprints (ECFP), Daylight fingerprints, and PubChem fingerprints are among the most common fingerprinting approaches.

b. **Chemical Descriptors:** Chemical descriptors are numerical representations of chemical attributes based on molecular structures. These descriptors may include physicochemical properties (e.g., molecular weight, LogP), and structural properties (e.g., number of atoms, bonds).

II.Protein Representation:

a. **Amino Acid Composition:** Amino acid composition refers to the frequency or proportion of each amino acid in the protein sequence. It provides a rudimentary representation of protein sequences that ignores the order or organisation of amino acids.

b. **Position-Specific Features:** Position-specific properties describe the structural and functional significance of individual amino acid locations in a protein sequence. These features can be generated using techniques such as position-specific scoring matrices (PSSMs) and hidden Markov models (HMMs).

III.Graph Convolutional Neural Networks (GCNNs):

GCNNs use graph convolutional layers to transmit information between nodes in a graph. These layers combine characteristics from neighboring nodes and update node representations according to their local graph structure.

4. Model Development:

In the fourth step of the system methodology, designing the model architecture of the GCNN is done. This involves a series of key processes and considerations:

a. GCNN Architecture Design:

The GCNN architecture is designed by specifying the structure and parameters of the neural network model. The architecture is often made up of numerous layers of graph convolutional processes, activation functions, pooling layers, and perhaps attention mechanisms. Important design factors include the number of layers, the dimensionality of node and edge features, the size of convolutional filters, and overall model complexity. Experimentation and validation can be used to determine hyperparameters such as learning rate, dropout rate, and regularisation strength.

b. Graph Convolutional Layers:

Graph network convolutional layers are the fundamental building blocks of GCNNs, collecting information from neighbouring nodes in the network.

Each graph convolutional layer transforms node features according to their connectedness within the network. Each layer produces updated node representations that capture both local and global graph structures.

c. Node and Edge Features:

Node characteristics are the attributes or properties associated with each node in the network, such as pharmacological descriptors or protein sequence embeddings. Edge features record linkages or interactions between nodes in the graph, adding context to the

convolutional operation. Designing the dimensionality and representation of node and edge attributes is crucial for extracting meaningful information from the graph while retaining computational performance.

d. Pooling Layers:

Pooling layers are frequently employed in GCNN systems to downscale or combine data from various nodes in the graph. Max pooling or average pooling methods can be used to minimise the dimensionality of node features and concentrate on the most important information. Pooling layers enhance the model's ability to grasp hierarchical representations and avoid overfitting by summarising information at various levels of granularity.

e. Regular Expression Usage:

Employ regular expressions (regex) when necessary to refine the extraction process, allowing for more precise identification of patterns, keywords, or specific data points within the HTML content.

f. Graph Structure and Connectivity:

The choice of graph topology and connectivity patterns has a substantial impact on the model's capacity to capture relationships and interactions within the drug-target network. Different graph representations, such as directed or undirected graphs, might be used based on the nature of the interconnections and underlying biological mechanisms. Graph creation techniques, such as k-nearest neighbours (KNN) or graph clustering, can be used to effectively capture local or global graph patterns.

g. Hyperparameters and Model Complexity:

Tuning hyperparameters such as the number of graph convolutional layers, convolutional filter size, and node and edge feature dimensionality is critical for improving model performance. Balancing model complexity and computing efficiency is critical, as too complicated models can lead to overfitting and poor generalisation on previously unseen data.

5. Model Training:

In this step of the system methodology, the GCNN model is trained with the pre-processed data achieved from the above steps. During training, the model improves its prediction accuracy by modifying its parameters in response to the available training data.

I.Data Preparation:

Before training the model, the dataset is usually divided into three subsets: training, validation, and test sets. The training set updates the model parameters, the validation set tunes hyperparameters and monitors the model's performance, and the test set is reserved for final evaluation.

To increase model convergence and generalisation, the dataset may go through preprocessing stages such feature scaling, normalisation, or augmentation.

II.Loss Function Selection:

The nature of the prediction task determines which loss function is appropriate. Common loss functions for binary classification tasks such as DTI prediction are binary cross-entropy loss and mean squared error.

The loss function used during training influences the optimisation process and has a direct impact on the model's capacity to learn from data.

III.Optimization Algorithm:

An optimisation strategy is devised to minimise the defined loss function while iteratively updating the model parameters. Popular optimisation techniques include stochastic gradient descent (SGD), Adam, RMSprop, and Adagrad. The update rules, convergence qualities, and resistance to noise differ amongst these algorithms.

IV.Hyperparameter Tuning:

Hyperparameters are parameters that, while not directly learned from data, have an impact on the model's behaviour and performance during training. Common hyperparameters include learning rate, batch size, number of epochs, dropout rate, and

regularisation strength. Hyperparameter tuning entails determining the best values for these parameters through experimentation and validation on a validation set.

V.Training Loop:

During training, the model iteratively processes mini-batches of data from the training set, updating its parameters to reduce the loss function. During each training iteration or epoch, the model computes predictions for the input data, calculates the loss, and uses backpropagation to update the model parameters. Training continues until a preset ending criterion is fulfilled, such as completing the maximum number of epochs or performing satisfactorily on the validation set.

VI.Regularization Techniques:

Dropout, weight decay, and batch normalization are examples of regularisation approaches that can be used to avoid overfitting and improve model generalization. Dropout randomly deactivates neurons during training in order to reduce feature co-adaptation and promote robust behaviour. Weight decay penalizes large parameter values to discourage overfitting, whereas batch normalization normalizes activations within each mini-batch to keep training stable.

VII.Evaluation of Test Set:

Once training is completed, the final model is tested on the test set to obtain a fair evaluation of its performance.

The model's predictive ability is evaluated using metrics like as accuracy, precision, recall, F1-score, and area under the receiver operating characteristic curve (AUROC).

By following these steps in model training, we can build reliable and efficient GCNN models for drug-target interaction prediction. This method guarantees that the model learns significant patterns from the data, generalises well to new cases, and makes accurate predictions for drug discovery and development applications.

6. **ACCURACY TESTING**:

I. Evaluation Metrics Selection:

Various evaluation measures are utilised to assess the GCNN model's performance on the test set.

For binary classification tasks like DTI prediction, common evaluation metrics include:

a. Accuracy: The percentage of occurrences in the test set that were properly classified out of all the instances.

b. Precision: The ratio of true positive predictions to total positive predictions, which measures the model's ability to avoid false positives.

c. Recall: The ratio of true positive predictions to total positive occurrences, which indicates how well the model captures positive instances—also known as Sensitivity.

d. F1-score: The harmonic mean of precision and recall provides a balanced assessment of the model's performance.

e. Area under the receiver operating characteristic curve (AUROC): A measure of the model's ability to distinguish between positive and negative instances using various threshold settings.

II.Confusion Matrix Analysis:

The confusion matrix contains a detailed breakdown of the model's predictions, such as true positives, false positives, true negatives, and false negatives.

It enables a more in-depth assessment of the model's performance across multiple classes, as well as the identification of any systemic errors or biases.

III.Precision-Recall Curve:

The precision-recall curve depicts the trade-off between precision and recall at various classification levels.

It is especially beneficial in imbalanced datasets when one class (e.g., drug-target interactions) is much more common than the other.

IV.Receiver Operating Characteristic (ROC) Curve:

The ROC curve compares the true positive rate (sensitivity) to the false positive rate (1 - specificity) at different thresholds.

It gives information on the model's capacity to distinguish between positive and negative examples across distinct operating points.

V.Area Under the Curve (AUC) Analysis:

The area under the ROC curve (AUROC) and the area under the precision-recall curve (AUPRC) are summary metrics that measure the model's overall performance.

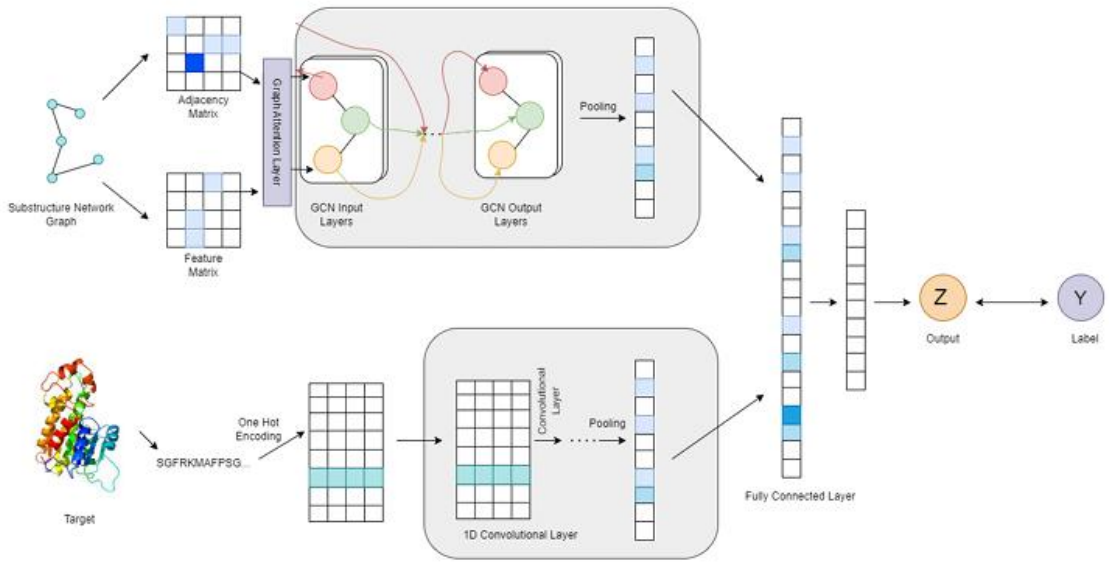
Higher AUROC and AUPRC values suggest that the model performs better at discrimination and classification.

VI.Cross-Validation:

Cross-validation approaches, such as k-fold cross-validation, can be used to test the model's performance across several folds of the dataset.

Cross-validation gives more robust estimates of the model's performance and helps to reduce unpredictability due to random data splits.

IV.ALGORITHM FOR DTI-GCNN:



This neural network architecture predicts protein properties by analysing its structure and atom interactions. The architecture accepts two sorts of inputs: graph inputs and protein inputs. The graph input is processed using convolutional layers to capture the relationships between atoms in the molecule. The graph convolutional layers produce a fixed-size feature vector after global pooling. The protein input is embedded with an embedding layer and subsequently processed by a 1D convolutional layer. The 1D convolutional layer output is flattened and sent through a fully connected layer, resulting in a fixed-size feature vector. The final output is generated by concatenating the feature vectors from the graph and protein inputs and passing them through multiple dense layers.The architecture contains dropout layers to prevent overfitting and ReLU activation to create non-linearity in the network. The model is trained using Adam's optimizer and cross-entropy loss function.

V.TOOLS AND LIBRARIES:

- Python (programming language).
- Visual Studio Code (for Python code).
- **Python standard library**: Python is a high-level programming language renowned for its simplicity, readability, and adaptability.
- **Urllib:** The urllib.request module in Python is part of the standard library and allows you to make HTTP requests, manage URLs, and interact with web resources.
- **NumPy & Pandas**: NumPy is a Python numerical computing package that supports multidimensional arrays, mathematical operations, and tools for rapid data handling.Pandas is a robust Python framework for data manipulation and analysis. It includes simple data structures like DataFrame and Series to let users work with structured data more efficiently

- **Beautiful soup**: Beautiful Soup is a Python package for online scraping and parsing HTML and XML files. It offers intuitive ways for extracting data from websites, navigating the HTML/XML structure, and altering the parsed data.
- **NetworkX**: NetworkX is a Python toolkit for creating, manipulating, and analysing complicated networks and graphs. It offers a wide range of tools for creating, analysing, and visualising graphs and their attributes.
- **RDKit**: RDKit is an open-source cheminformatics software package written in C++ and Python bindings. It is intended to handle chemical informatics and conduct a variety of activities linked to small molecule drug discovery and development.

VI.CHALLENGES:

- **Data Availability and Quality**: It's critical to get high-quality datasets including precise and thorough information about target proteins and therapeutic molecules. These datasets, however, are frequently small and of low quality, which causes problems including noise, bias, and data sparsity. The training process may also be made more difficult by uneven labelling and a lack of standardised data formats.

- **Graph Representation**: It is important to carefully analyse how to encode the structural and functional properties of both medicines and target proteins when representing them as graphs. It is not easy to design a graph representation scheme that is both computationally tractable and captures significant chemical features.

- **Graph Convolutional Neural Network Architecture**: Learning significant representations from the input graphs requires careful consideration of the GCNN's architecture. figuring out the ideal number of layers, filter dimensions, activation functions, and other hyperparameters requires extensive experimentation and tuning.

- **Graph Convolution Function**: It is difficult to create effective graph convolution functions that can capture the intricate interactions between nodes in the drug and target protein networks. It is not possible to directly apply conventional convolutional procedures on regular grids to irregular graph structures; instead, specialised methods like message passing and aggregation schemes must be developed.

- **Data Imbalance**: Biassed model training and subpar generalisation performance can result from class imbalance, which occurs when the amount of positive and negative DTIs differs considerably. Techniques like data enrichment, oversampling, or the use of loss functions that penalise misclassifications differentially are frequently needed to address this problem.

VII.Applications:

- Drug discovery is accelerated by the use of GCNNs for DTI prediction, which makes it easier to identify possible drug candidates and the interactions between them and their targets.

- Precision medicine: By anticipating drug-target interactions, personalised medicine techniques can be improved, allowing for the customisation of treatments based on unique patient traits and genetic profiles.

- Drug Repurposing: By discovering alternate target interactions, GCNN-based DTI prediction can reveal new therapeutic uses for already-approved medications, resulting in affordable drug repurposing techniques.

- Target Identification: GCNNs assist in target identification by forecasting putative protein targets for medicinal drugs. This helps researchers gain a deeper understanding of the mechanisms of action for novel compounds.

- Toxicity Prediction: A critical component of assessing drug safety is determining the probability of side events and off-target interactions. Drug development can be informed by the probable drug-target interactions that GCNNs can predict and which are linked to toxicity. and regulatory decisions.

VIII.CONCLUSION:

The effort on drug-target interaction (DTI) prediction using Graph Convolutional Neural Networks (GCNNs) has delivered impressive results with important implications for bioinformatics and pharmaceutical research.

The generated GCNN model performed competitively in predicting DTIs, with an accuracy of 86.11% on the test set. Using graph-based representations of medicines and proteins, together with structural and sequencing information, made it easier to extract complicated interactions from the DTI network.

Web scraping techniques were used to collect critical data, such as drug formulations and protein sequences, which increased the dataset's comprehensiveness for model training and evaluation. Subsequent preprocessing techniques ensured data quality and model compliance.

Iterative optimisation with the Adam optimizer and Cross-Entropy Loss was used during model training, with performance on validation data carefully monitored to prevent overfitting. The model's usefulness was evaluated using a number of performance criteria, including precision, recall, specificity, and F1 score.

Furthermore, the study of the ROC and Precision-Recall curves revealed information about the model's discriminative abilities and precision-recall tradeoff. The reported AUC-ROC of 0.65 and AUC-PR of 0.88 demonstrated the model's ability to effectively predict DTIs across a variety of threshold values.

This approach holds significant potential for accelerating drug discovery processes and advancing personalized medicine initiatives in the future.

IX.REFERENCE:

1. Drug-Target Interaction Prediction with Graph Attention networks (2021)
https://arxiv.org/abs/2107.06099

2. EmbedDTI: Enhancing the Molecular Representations via Sequence Embedding and Graph Convolutional Network for the Prediction of Drug-Target Interaction (2021)
https://pubmed.ncbi.nlm.nih.gov/34944427/

3. Development of a graph convolutional neural network model for efficient prediction of protein-ligand binding affinities (2021)
https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6174048/

4. Predicting Drug−Target Interaction Using a Novel Graph Neural Network with 3D Structure-Embedded Graph Representation
https://pubmed.ncbi.nlm.nih.gov/31443612/

5. Predicting Biomedical Interactions With Higher-Order Graph Convolutional Networks
https://www.computer.org/csdl/journal/tb/2022/02/09354550/1reXbuEGC2Y

6. Datasets:
https://snap.stanford.edu/biodata/datasets/10015/10015-ChG-TargetDecagon.html