UNIVERSITY OF CAMBRIDGE
Department of Computer Science and Technology

Research project report title page

Candidate **2110K**

"Optimising representation learning of heterogeneous cancer data: how a computer scientist could fight cancer"

Submitted in partial fulfilment of the requirements for the Master of Philosophy in Advanced Computer Science

Graduate Education Office, Room FS03
Department of Computer Science and Technology, William Gates Building, 15 JJ Thomson Avenue, Cambridge, CB3 0FD

Total word count: 10,577

Abstract

Motivation

Cancer is the leading cause of death worldwide, and the identification of new drugs and risk factors for cancers are time-consuming. Fortunately, the abundance of biological data enables the use of computational methods to accelerate these processes. Traditional computational models for cancers do not differentiate between different types of cancers and are often task-specific. As a result, these models lack generality and cannot use all the available associations in biological systems.

Result

This project develops BIO-RGCN, an extendable framework to predict the associations between chemicals and cancers. BIO-RGCN can learn node representations in heterogeneous networks and predict the existence of links between nodes. This framework has two advantages: first, BIO-RGCN is general; it is applicable to a wide range of link prediction tasks on heterogeneous networks. Secondly, it addresses the unbalance problem of networks by decomposing large networks into smaller chunks.

One technology used in this project is graph neural networks, which can efficiently aggregate information on network data. Besides, natural language processing is used to generate embeddings for chemicals and cancers, which enables the model to deal with unseen inputs.

In addition to the chemical-cancer link prediction, BIO-RGCN is also used to predict drug target interactions. Evaluations of the framework on different tasks demonstrate its stability in link prediction tasks; the outputs from the model are consistent with existing medical literature.

Implementation

A demonstration of prediction results can be accessed through the following link[1], which is a self-contained jupyter notebook.

[1] `https://colab.research.google.com/drive/18ZTZYMXKOGT-xtpKHir11QznWhfWomZG?usp=sharing`

Contents

Chapter 1

Introduction

Cancer has always been the leading cause of death around the world; according to the annual report [1] from cancer research UK, approximately 990 people are diagnosed with cancer on a daily basis. What lies in the centre of the treatment of cancer and other genetic diseases, are targeted therapy and prevention [2, 3]. However, the development of new drugs and the identification of risk factors heavily rely on clinical experiments, which are time-consuming. The use of computational approaches, on the other hand, can often take advantage of the existing data and accelerate the processes of drug development and risk factor identification.

This project aims to develop a computational framework, BIO-RGCN, for integrating biological networks. This framework is designed for the link prediction task on heterogeneous networks; it primarily addresses the problem of imbalance of network data set, facilitating data integration process. This framework can be used for multiple tasks, including establishing chemical-disease associations and predicting drug target-protein interactions (DTIs).

BIO-RGCN uses Graph Neural Networks (GNNs) as the building block. GNNs are neural networks designed for graphical data; they can efficiently extract information from the network data and create embeddings for every node of the network. The resulting embeddings can then be applied to

standard tasks such as link prediction (predicting for the existence of edges between nodes) and node classification (classifying the type of nodes). In particular, BIO-RGCN makes uses of two types of GNNs, Graph Convolutional Networks [4] and Relational Graph Convolutional Networks [5], to integrate information from heterogeneous networks.

For the establishing chemical-cancer association task, this work creates a customized dataset called CA-CHEM to model the direct relations between various types of cancers and chemicals. Each chemical is associated with one type of cancer either as a substance to treat cancer or a biomarker, which could be a potential risk factor leading to cancers. For the DTIs prediction task, on the other hand, I use an existing dataset to compare the performance of BIO-RGCN with other machine learning models. In addition, natural language processing (NLP) tools are exploited to improve the performance of BIO-RGCN framework on the chemical-cancer link prediction task. The information is extracted from biomedical text in the format of embeddings. The inclusion of NLP components enables the system to generalize to unseen chemicals and cancers.

Figure 1.1 provides an overview of BIO-RGCN framework on the chemical-cancer association prediction task; it presents the general scope of this project and the overall structure of BIO-RGCN framework. From left to right, different types of data (text, gene-gene and pathway-pathway interactions information) are converted to their vector presentations (embeddings), these embeddings are then used as the input of relational graph neural networks to establish the association between chemical and disease nodes. The output of the system will indicate the existence of a potential association between two nodes.

Overall, my work has resulted in the following contributions:

1. An extendable framework for the prediction of links between diseases and chemicals, using a Graph Convolutional Network for data integration, with evidence from an unusual number of multiple sources of heterogeneous type.

2. A neural architecture implementing the above framework, representing the first solution to the problem that the set sizes of evidence from different sources can be extremely unbalanced; including an evaluation of the robustness of the framework, comparing against baselines.

3. Exploration of the best representation for genes and pathways (embeddings vs. one-hot).

4. Exploration of the best representation of chemicals and diseases in an NLP context, using Bidirectional Encoder Representations (BERT).

5. A demonstration of the generality of the framework on drug target interactions prediction task with corresponding evaluation.

6. Creation of a new dataset consisting of 15 types of cancers, 1020 types of chemicals, 2178 pathways, and 18009 types of genes, which enables the evaluation of the system.

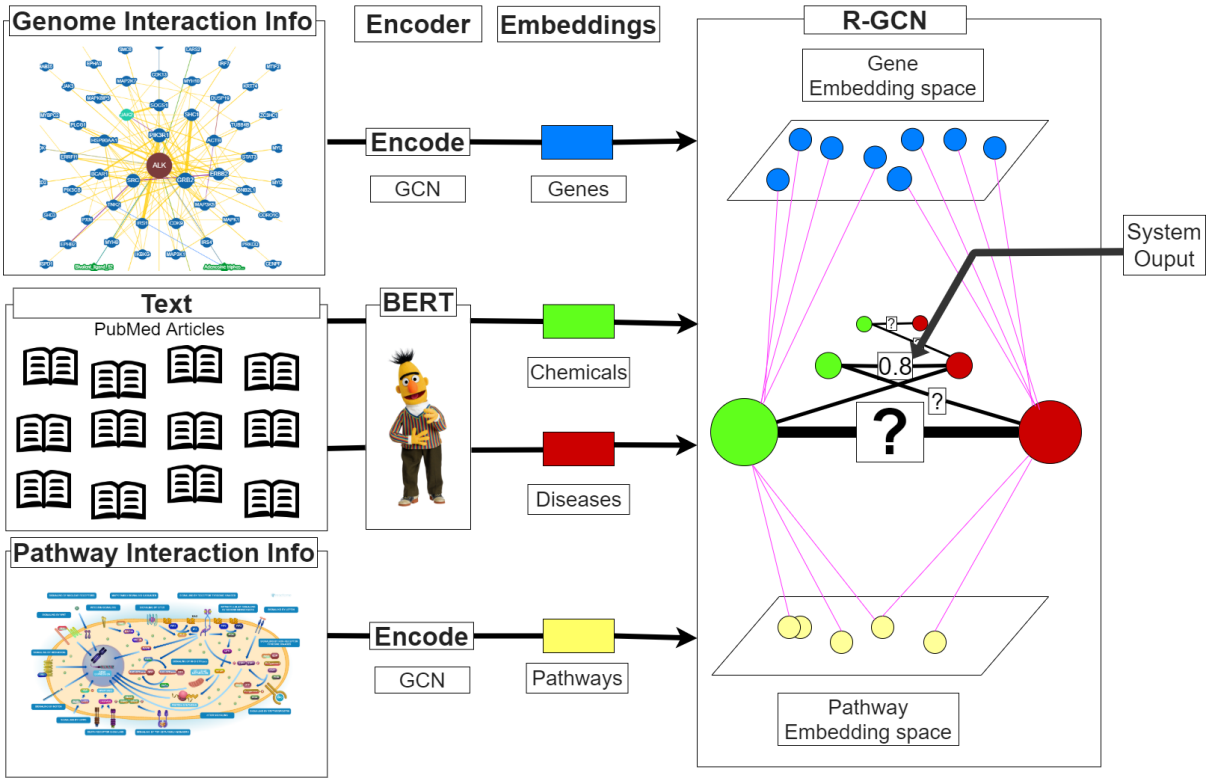

Figure 1.1: An overview of BIO-RGCN framework on the chemical-diseases prediction task. From left to right, genome interaction, pathway interaction and text data (Pubmed articles) are propagated through neural models to create embeddings. These embeddings are then integrated into the multi-modal graph to facilitate the link prediction task. The output of the system will indicate if there is a link between a chemical and a disease node.

Chapter 2

Related work

This chapter reviews prior works on applying NLP technologies and machine learning to biomedical science. These works provide inspiration for this project, motivating the use of a combination of NLP and computational methods on biological networks and cancer research.

2.1 Machine learning for network biology

Biological networks are used to model complex biological systems, where nodes of networks can represent a wide range of biological units, such as proteins, genes, and ecosystems. It is important to perform analysis on the biological networks because these analyses may reveal undiscovered interactions in the original biological systems.

A great number of approaches have been proposed to analyze biological networks with machine learning models over the years, providing insights in laboratory experiments.

Drug repositioning

Veselkov [7] combined support-vector machines (SVMs) and unsupervised learning algorithms to predict the cancer-beating molecules based on gene-gene and gene-drug interactions information. In this work, se-

lected molecules are assigned scores by the SVM classifier to show their potentials to beat cancers. While this work innovatively combines information from two networks, it does not differentiate between different types of cancer. This fact restricts the use of this work for guiding more detailed cancer research.

Drug Pair Side-effects prediction

Another work which makes use of multimodal graph data is Decagon[8]; it predicts polypharmacy side effects basing on drug-drug, gene-gene, and drug-gene interactions. It relies on the convolutional neural network (GCN) model to effectively encode the network information of the multimodal graph; however, this model does not apply to the situation where the number of various types of nodes is unbalanced.

Drug-target interaction (DTIs) prediction

Traditional DTIs prediction relies on the experiments and 3-d structures of drugs and target proteins [9]. However, the use of heterogeneous data sources and computational methods could be a cost-effective way to discover unprecedented interactions. Luo [10] proposes a computational framework called DTINet to predict DTIs based on drug-drug, protein-protein, drug-disease, and other interactions. Despite the success of its method to integrate heterogeneous networks, there is still space for further improve by the use of graph auto-encoder (GAE) [11].

While the research question of above projects varies, they all follow the identical pipeline: different biological networks are firstly aggregated into a multimodal network, and then machine learning algorithms come to plays a role in extracting information and making predictions. The challenge of applying ML to biological network often lies on creating effective architecture to filter out the noises in the data integration process [12].

In this thesis, a new framework is proposed to aggregate information from heterogeneous networks and make link prediction on a multimodal graph.

2.2 Biomedical text mining

Biological networks are powerful, but the creation of network dataset from billions of biomedical literature requires great efforts. This is where biomedical text mining come to play a role.

With the success of pre-trained language models in the general NLP field, several pre-trained language models are created specifically for biomedical text mining; for example, BioBERT [13] is trained mainly on PMC full-text articles, and this model is used for downstream tasks such as named entity recognition and question answering [14, 15]. Another pre-trained language model for the biomedical usage is SciBERT, which is trained on a mixture of biomedical text and computer science papers, resulting in a slightly lower performance than bioBERT on biomedical text mining [16].

While pre-trained models like BioBERT improve the performance of standard text mining tasks, rare work has been done in applying language models to computational biology. In this project, the pre-trained language model will be combined with the biological network to make predictions about unseen relations between biological entities.

On the other hand, the traditional biomedical text mining study focuses on building tools to facilitate the annotations. PTC [17] and PubTator [18] are two popular tools for extracting biological entities from PubMed. Pyysalo [19] creates LION LBD, a literature-based discovery system, to extract information from cancer-related literature and construct biological networks. LION LBD uses named entity recognition and co-occurrence based metrics to identify links between chemicals, diseases, and other biological entities. All these tools mentioned above curate existing information from biomedical literature, but it is not capable of discovering new links between biological entities.

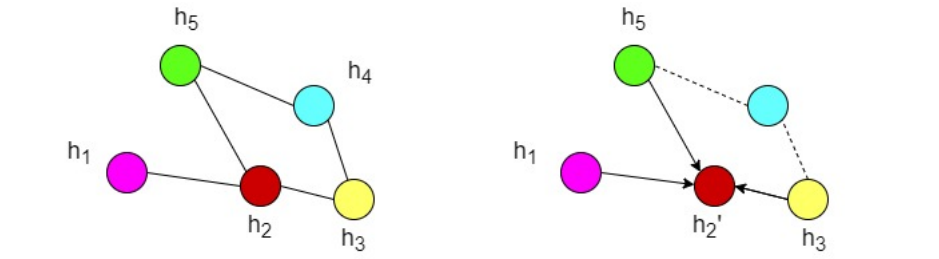

Figure 2.1: An illustration of GNNs. (a) Every node in the graph is associated with a vector h_i. (b) At the training stage, the vector representations of every node are updated with the vectors of neighbourhood nodes.

2.3 Technical background

This section will provide background on two fundamental tools used by this project: graph neural networks (GNNs) and pre-trained language models.

2.3.1 Graph neural networks

Graph neural networks (GNNs) are a type of neural networks for modelling graph-structured data. Graph data is non-Euclidean, so traditional neural models such as convolutional neural networks [20] cannot be applied. GNNs operates on a graph by aggregating information from neighbour nodes. The earliest work about GNNs dates back to 2005 [39] when the name of graph neural networks is given. The most recent work extends GNNs in various ways: graph attention networks [40] improves upon GCNs by given the node ability to specify distinctive weights to neighbourhood nodes; relational GCNs [5] operate on multi-relational data with efficient regularization schemes to avoid overfitting. The formal definition of GNNs is given as the following.

Given a graph G = (E,V), where E represents a set of edges for G, and V is a set of vertices. Every node $i \in V$ is then associated with a vector representation h_i, and every edge $(i, j) \in V$ is associated with a vector $h_{i,j}$. At the training stage (so-called message passing stage), the vector presentation

for edges and nodes will be updated to encode the structure of graph G.

Figure 2.1 shows the training process of the a simple GNN consisting of five nodes. Five nodes are firstly initialized with vector $h_1, \ldots, h_5$. At the message passing stage, node vector h_i is updated according to message passing function-2.1, resulting in a new representation h'_i for node i.

$$h'_i = \gamma(\underset{j \in \mathcal{N}(i)}{f_{agg}} \{\phi(h_i, h_j, h_{ij})\}) \tag{2.1}$$

where f_{agg} is the aggregation function, it can be sum, average, or max operation. γ and ϕ are message and update functions, respectively. $\mathcal{N}(i)$ represents the set of nodes which are directly connected to node i. Noticeably, this formula define the most general form of GNNs [21], different combination of f_{agg}, γ, and ϕ result in various types of GNNs. For instance, graph convolutional networks (GCNs) [4] use ***sum*** as the aggregation function, its message passing function is defined according to equation-2.2.

$$h'_i = \sum_{j \in \mathcal{N}(i)} \{\frac{1}{\sqrt{deg(i)}\sqrt{deg(j)}} W h_j\} \tag{2.2}$$

where $deg(i)$ represents the degree of node i in the graph, i.e., the number of edges directly connected to node i.

2.3.2 Pre-trained language model

In transfer learning, a neural network is first trained on a general dataset, and then the resulting network can be fine-tuned on target dataset. The networks trained in this manner usually perform better than the model trained only on the target dataset. This is because transfer learning enables the transfer of knowledge from the general domain to a specific problem. In the computer vision field: Yosinski et al. show that deep learning models pre-trained on ImageNet can have a better performance on image classification task with other datasets [22, 26].

In the context of natural language processing (NLP), most of the NLP tasks

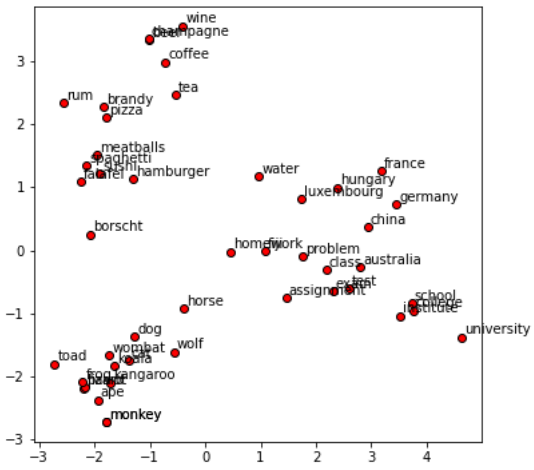

Figure 2.2: A visualisation of word embeddings. The words with similar meanings are projected to closer locations in the embedding space.

require the common knowledge of the language, and transfer learning on large text corpus can encode the syntactic and semantic knowledge of language into the model. As a result, transfer learning can often significantly improve the performance of models in various tasks, such as text classification, question answering, and machine translation [23, 24, 25]. Another reason why transfer learning is popular in NLP is the size of unlabeled text data: transfer learning can be an effective way to make use of these unlabeled data. The term, pre-trained language model (PLM), usually refers to neural models that are trained on the unlabeled text; these models can be applied to different downstream NLP tasks to improve the performance.

Early PLMs focus on creating embeddings at word level: word2vec, GloVe, and fastText can all create word embeddings with unlabeled data [27, 28, 29]. Figure 2.2 visualize the word embeddings for some common words. These embeddings capture the semantic meanings of words and project them to the 2d space. These word vectors capture the co-occurrence of words in the text, and they are often used as the input for other neural architectures. The use of word vectors greatly improve the performance of neural models; however,

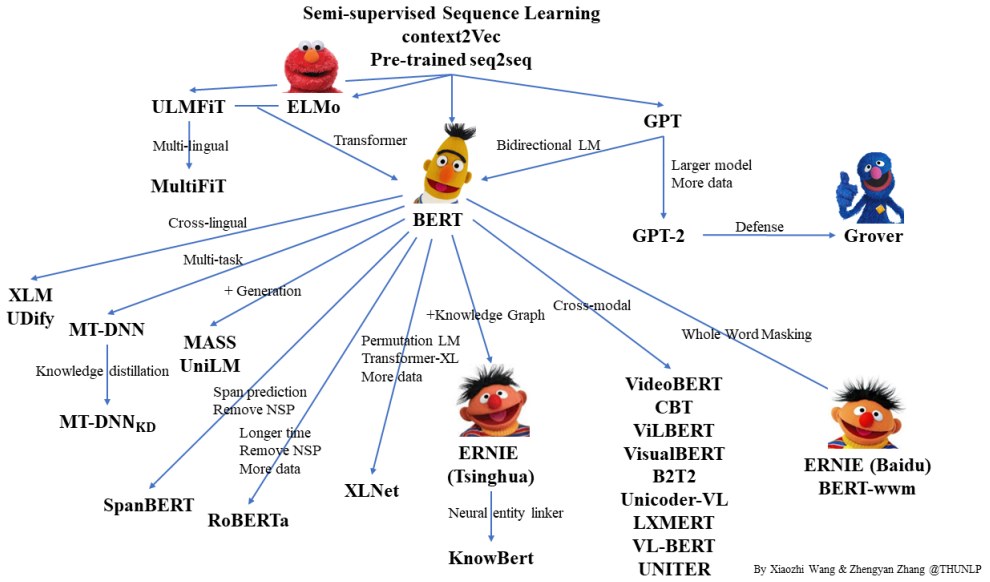

Figure 2.3: Relationships of pre-trained language models. BERT plays an important role in connecting different models. From thunlp[38]

the information encoded by word embeddings are limited because they poorly capture the semantic meaning of the same word in different contexts. One word can have different meanings in different contexts, while the word vector approach maps every word to one vector.

In order to account for this problem, a series of approaches are proposed to create ***contextual word embeddings***. Figure 2.3 shows the development of these approaches. Context2Vec, ELMo, and ULMFiT are early works of contextual language embeddings; these models mainly use recurrent neural networks as the building blocks [30, 31, 32]. They take a sequence of words (sentences) as the input and output contextual embeddings for each word in the input sequence. In what follows, Bidirectional Encoder Representations from Transformers (BERT) are proposed by Devlin et al. [33]. BERT achieves state-of-the-art performance on eleven NLP tasks and motivates a large number of later works, as shown in figure 2.3.

BERT is different from previous works in two ways: first, it uses transformers [34] as building blocks for the model. The transformer is better at capture

long-distance relations in the sentence while compatible with parallelization of computing. Secondly, the amount of training data BERT uses is unprecedented. Unsupervised training makes it possible for BERT to use $3,300$ million words of plain text in the pre-training process. Follow-up works of BERT, such as XLNet, RoBERTa, and GPT-2, are all transformer-based models with similar size of training data as BERT [35, 36, 37].

Chapter 3

A new model for predicting chemical/cancer interaction

In this chapter, the design and implementation details of BIO-RGCN will be described. Since BIO-RGCN is designed for chemical/cancer interaction prediction, this chapter will start by introducing the task of link prediction on the customized dataset. Following that, the architecture and training regime of BIO-RGCN will be detailed. Finally, the chapter ends with an approach to create embeddings for nodes in the network from the NLP perspective.

3.1 Overview of the task

To investigate the association between cancers and chemicals, a dataset called CA-CHEM is created from CTD and BioGRID[1] [41, 42]. It consists of 15 common types of cancer, 1020 types of chemicals, 18009 genes, and 2178 pathways. Basic statistics of the dataset is shown in table-3.1.

Figure 3.1 illustrate the structure of dataset. The task is to predict the relations between cancer nodes and chemical nodes with all the additional information (pathway components and gene components).

[1]Comparative Toxicogenomics Database (CTD) and BioGRID are two publicaly available databases.

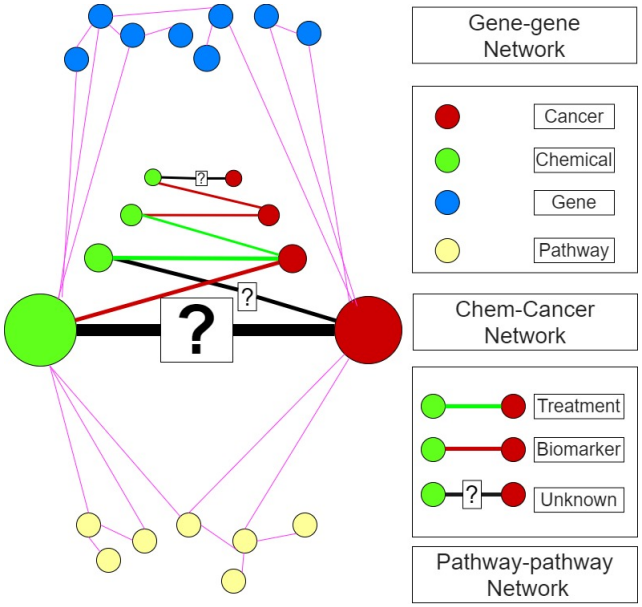

Figure 3.1: Overview of CA-CHEM dataset. It consists of 4 types of nodes. There are two types of links between chemical and cancer nodes: treatment link indicates that a chemical has a treatment effect on a type of cancer while biomarker represents that exposure to a chemical may cause some diseases. The task is to establish the association between cancer nodes and chemical nodes with unknown link type.

	Dataset statistics
Nodes	21,182
Cancer Nodes	15
Chemical Nodes	1020
Gene Nodes	18,009
Pathway Nodes	2,178
Edges	1,098,382
Gene-Gene Edges	367,205
Gene-Chem Edges	409,154
Gene-Cancer Edges	2,021
Chem-Cancer Edges	1,794
Pathway-Chem Edges	256,495
Pathway-Cancer Edges	40,223
Pathway-Pathway Edges	21,490

Table 3.1: Statistics for CA-CEHM networks

.

The main challenge of this task is that the number of chemical and cancer nodes are much smaller than the gene and pathway nodes. In the next section, we will address this problem using BIO-RGCN model.

3.2 BIO-RGCN

Predicting the cancer-chemical association on the CA-CHEM dataset can be modeled as a link prediction problem on a multimodal graph. Suppose graph G= (V,E) , where V is the vertices set for G and E is the edge set for G. Every node $i \in V$ in the graph is associated with a vector x_i while every edges $(i, j) \in E$ is associated with a real number r indicating the type of the edge.

BIO-RGCN decomposes the multimodal graph into two parts: bipartite graph G_{bipart} and additional graph $G_{addtional}$ as shown in figure 3.2. Formally, graph $G = G_{bipart} \cup G_{additional} = \{V_{bipart}, E_{bipart}\} \cup \{V_{addtional}, E_{addtional}\}$, where $G_{additional}$ includes gene-gene and pathway-pathway networks and G_{bipart} includes all the chemical nodes, cancer nodes, and all the nodes that are directly linked to chemicals and different types of cancer.

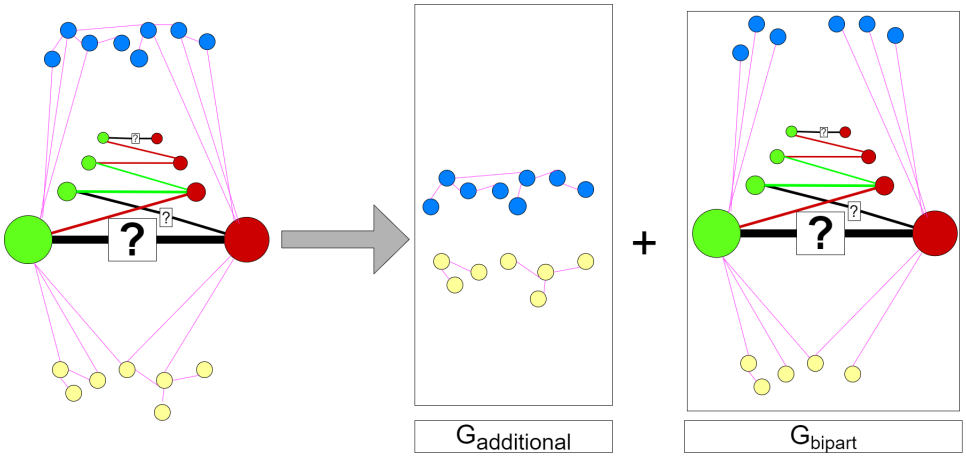

Figure 3.2: Decomposition of CA-CHEM graph. The heterogeneous networks are decomposed into two parts: $G_{additional}$ includes additional information, which are gene-gene and pathway-pathway interactions. G_{bipart} includes all the nodes that are directly connected with target nodes (chemicals and cancers).

Bio-RGCN model will take a chemical node x_i and a cancer node $x_j \in V_{bipart}$ as the inputs and predict the edge type of (i, j). To accomplish this goal, BIO-RGCN has three components.

- **GCN encoder:** It learns embeddings for all genes and pathways nodes in $G_{addition}$ using graph auto-encoder [43].
- **R-GCN encoder for bipartite graph:** Using embeddings of genes and pathways nodes, it generates embeddings for chemical and cancer nodes in G_{bipart} which we need to make prediction for.
- **Decoder:** It uses the embeddings of chemicals and cancers in G_{bipart} to produce a real number, indicating the type of edge (i, j) for chemical i and cancer j.

3.2.1 Learning gene/pathway representations with the GCN encoder

Encoder

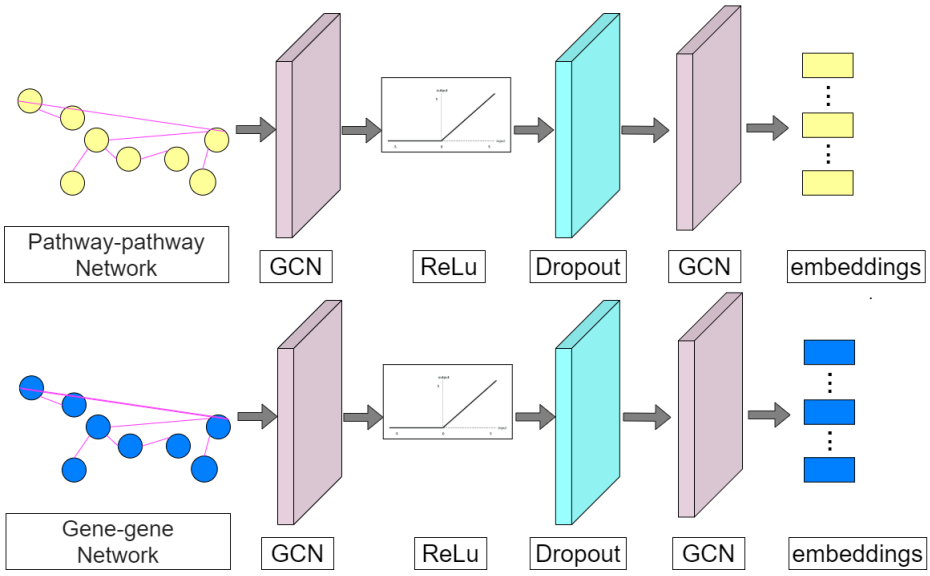

Figure 3.3: The GCN encoder structure to learn the representation of genes and pathways. Genes and pathways are trained independently. From left to right, the gene-gene networks and pathway-pathway networks with one-hot embeddings go through two layers of GCNs with a non-linear layer and a dropout layer. The final outputs are vector representations (embeddings) for every node in two networks.

Graph convolutional network(GCN) encoder will learn representations for gene and pathway nodes basing on gene-gene network and pathway-pathway network (figure 3.3). Firstly, a GCN layer is defined in equation-3.1, which takes a node vector $x_i \in \mathbb{R}^n$ as the input.

$$GCN(x_i) = \sum_{j \in \mathcal{N}(i)} \frac{1}{\sqrt{deg(i)}\sqrt{deg(j)}} W x_j \tag{3.1}$$

where $\mathcal{N}(i)$ is the set of nodes which are directly connected to node i; $W \in \mathbb{R}^{n \times m}$ is the weight matrix; $deg(i)$ is the number of incoming edges for node i and x_j is the vector representation for node j.

The GCN encoder used in this section consists of two layers of GCNs, a non-linear layer , and a dropout layer as shown in equation-3.2.

$$z_i = encoder(x_i) = GCN(Dropout(ReLU(GCN(x_i)))) \tag{3.2}$$

where vector $z_i \in \mathbb{R}^m$ represents the output of the encoder. It is the resulting embeddings of pathways or genes.

Learning

With the output from the encoder, we still need to define a loss function to learn the representation of gene/pathway nodes, which is defined as the following:

$$C = -\sum_{(i,j) \in E} \log(sigmoid(z_i z_j)) - \sum_{(m,n) \notin E} \log(1 - sigmoid(z_m z_n)) \tag{3.3}$$

Where E represents the edge set for gene-gene network or pathway-pathway network. (m, n) are sampled negative examples. By minimizing this cost function C, we can obtain a set Z that includes all the vector representations of genes/pathways. These vectors implicitly encode the information of gene-gene/pathway-pathway networks. *Sigmoid* function is used to project the

probability to $(0, 1)$ [2].

3.2.2 R-GCN encoder for bipartite graph

R-GCN encoder will take the generated embeddings of genes and pathways as the inputs, propagating them through bipartite graph G_{bipart} to get vector representations for chemical and cancer nodes.

Relational graph convolutional networks (R-GCNs) are proposed by Schlichtkrull to model the relations in the knowledge graph [5]. It has a similar structure as GCNs while accounting for various types of edge between different nodes. Assuming that there are $|R|$ different types of relations and $x_i \in \mathbb{R}^n$, R-GCN is defined as the following:

$$\text{R-GCN}(x_i) = \sum_{r \in R} \sum_{j \in \mathcal{N}_r(i)} \frac{1}{\sqrt{|\mathcal{N}_r(i)|}} W_r x_j + W_0 x_i \tag{3.4}$$

Where $\mathcal{N}_r(i)$ is the set of nodes which are directly connected to node i under relation r; $W_r \in \mathbb{R}^{n \times m}$ is the weight matrix for relation r, and $W_0 \in \mathbb{R}^{n \times m}$ is the weight matrix modeling the self-loop. Different from GCNs layer, R-GCNs layer maintains multiple weight matrices $W_0, W_1, \ldots, W_{|R|}$ to model different types of relations.

R-GCN encoder consists of multiple linear layers, two layers of R-GCNs, one non-linear layer and one dropout layer, as shown in equation-3.5.

$$z_i = encoder(x_i) = \text{R-GCN}(Dropout(ReLU(\text{R-GCN}(Linear(x_i))))) \tag{3.5}$$

Figure 3.4 gives a graphical representation of R-GCN encoder. The inputs of decoders will be the one-hot encoding of chemical and cancer nodes, together with the learned representation of genes and pathways. In order to incorporate generated embeddings of genes and pathways from GCN encoder, **linear layers** are used to force different types of node vectors to have the

[2]Sigmoid function: $y(x) = \frac{1}{1+e^{-x}}$. It is used to project the vector distance to probability.

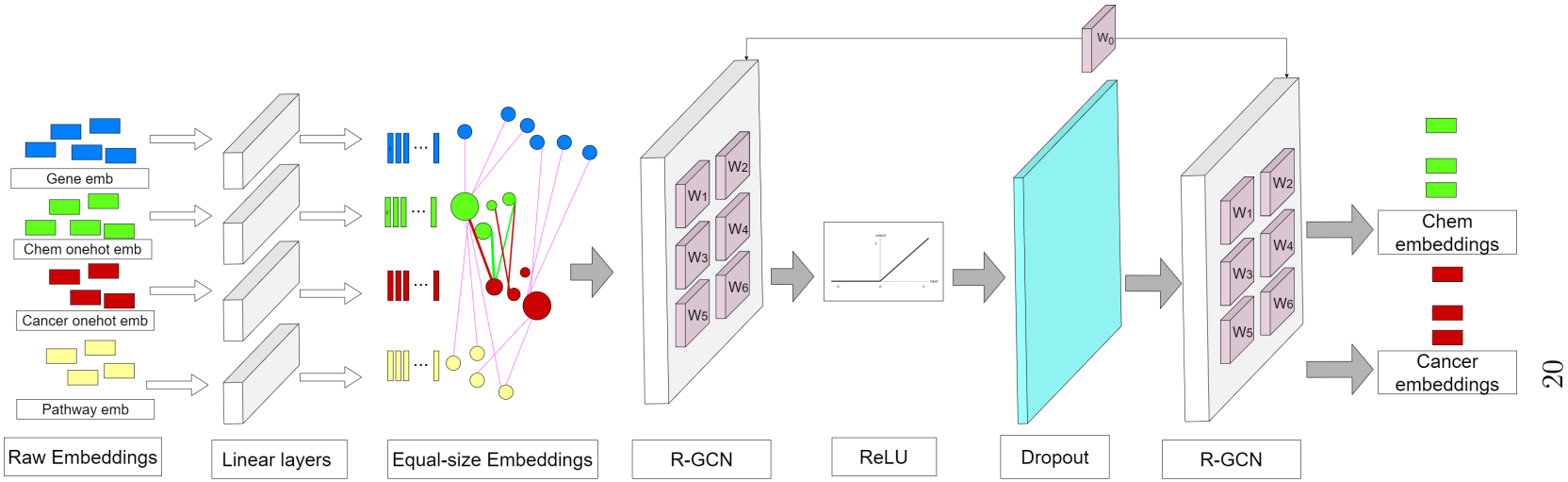

Figure 3.4: The R- GCN encoder to learn the representation of chemicals and cancers. From left to right, there are four sets of inputs: one-hot embeddings for cancers, one-hot embeddings for chemicals, and gene embeddings (emd) and pathway embeddings (emd) from the GCN encoder. These inputs first go through linear layers to be transformed to the same size, and then they are used to initialize nodes of G_{bipart}. Finally G_{bipart} go through R-GCN encoder, which consists of two R-GCN layers, a non-linear layer and a dropout layer. The output will be two sets of embeddings for chemicals and cancers. These two sets of embeddings will then be used to make prediction.

same dimension. Following that, four sets of embeddings with equal size are propagated through R-GCN layers and a non-linear layers to obtain the embeddings of chemicals and cancers. There are six types of edges, and therefore there are six weight matrices in each R-GCN layer; self-loop weight matrix is shared between two R-GCN layers.

3.2.3 Decoder

The decoder will take a tuple of learned representations for chemicals and cancers (z_i, z_j) as the input and compute the probability distribution of over possible edge types of (i, j). Equation-3.6 define the decoder for BIO-RGCN.

$$p_r(i, j) = sigmoid(z_i^T M_r z_j) \tag{3.6}$$

where $z_i, z_j \in \mathbb{R}^m$ are the encoding vectors for chemical and cancer nodes. $M_r \in \mathbb{R}^{m \times m}$ is the weight matrix encoding the interactions between two vectors for edge type r. $p_r(i, j)$ represents the probability of edge (i, j) being type r.

Data imbalance problem

The next step to establish chemical-cancer association will be designing a proper cost function, and to accomplish this goal, we need to resolve the data imbalance problem.

The data imbalance problem exists because the number of cancer and chemical nodes is much smaller than the number of gene and pathway nodes. Furthermore, there are six types of edges in the bipartite graph, but we are only interested in two of them: chemical-caner (treatment) and chemical-cancer (biomarker).

In the traditional link prediction approach, the cost function will include all the edge types; however, in this imbalanced graph, including all the edge type in the cost function will result in a poor performance in predicting links we are concerned about since chemical-cancer edge only accounts for 0.2% of total edges.

Cost function

To solve the data imbalance problem, we enforce a constraint on the cost function such that it only includes the edges we are interested in. Let E_{CC} represent the edge set containing all the interactions between chemicals and cancer in the CA-CHEM dataset; $E^{\complement}_{CC}$ includes the same number of negative sampled edges. Negative sampling is achieved by randomly choosing (chemical,cancer) pairs that does not existing in E_{CC}.

The cost function C is defined in equation-3.7.

$$C = \sum_{(i,j)\in E_{CC}} -\log(p_r(i,j)) - \sum_{(m,n)\in E^{\complement}_{CC}} \log(1 - p_r(m,n)) \tag{3.7}$$

By minimizing this cost function, we will be able to obtain parameters for R-GCN encoder and decoder and compute the type of association for any (chemical, cancer) tuples.

3.3 Learning NLP embeddings for chemicals and diseases with PLMs

In the last section, we have defined a framework, BIO-RGCN, to make predictions on heterogeneous networks. However, there is still space for improvement. In the original system, I combine learned embeddings of genes/pathways with **one-hot encoding** of chemicals and cancers to make a prediction for the chemical-cancer association. The problem is that **one-hot** encoding is not the best way to represent cancer and chemical nodes. This system can be enhanced by the use of pre-trained language models (PLMs), which can create embeddings for cancer and chemical nodes.

In what follows, an approach to generate embeddings of chemicals and cancers with a pre-trained language model - BERT - will be presented.

Contextual word embeddings from BERT

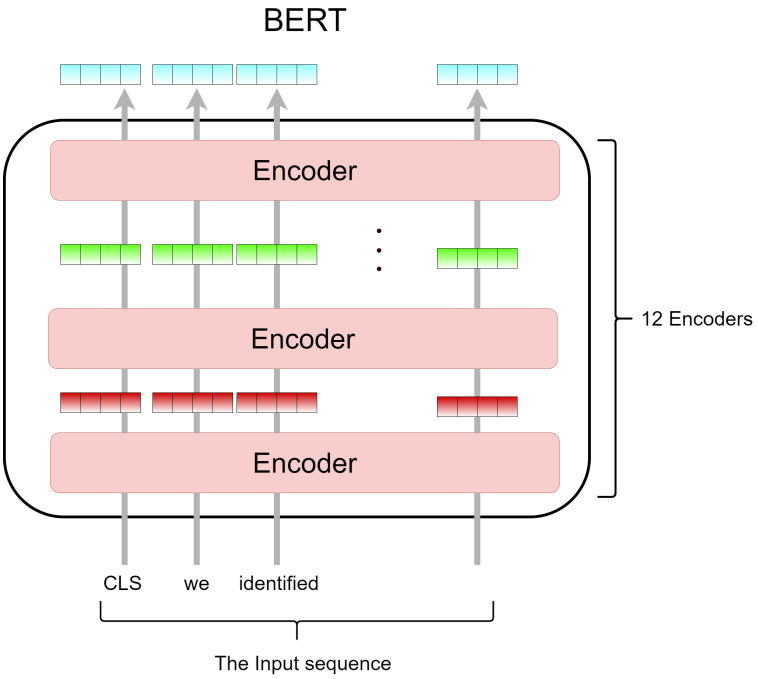

Figure 3.5: The architecture of BERT: The input sentences"we identify ..." go through BERT model (from the bottom to the top). The blue, green, and red squares represent outputs vector representations of input sequences from different layers. There are 12 encoders, and all encoders are implemented with transformer. "CLS" is a special token to indicate the start of an input sequence.

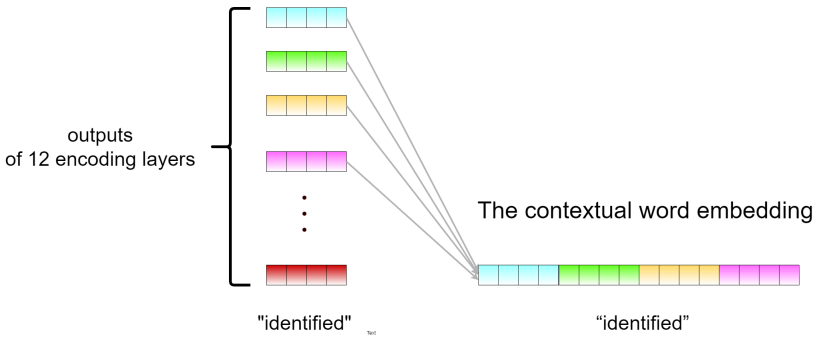

Figure 3.6: The creation of embeddings from BERT: For a single word “identified”, the contextual word embedding for it can be created by concatenating the last four layers of BERT encoders.

Bidirectional Encoder Representations from Transformers (BERT) is a powerful transformer-based model [33]. It can be applied to a wide range of downstream NLP tasks, such as text classification, question answering, and named entity recognition. Furthermore, BERT can be used to create contextual word embeddings without any fine-tuning. Figure 3.5 shows the architecture of BERT, it consists of 12 layers of transformer encoders, and each layer of encoder will create vector represents for all the input words. There are multiple ways to get contextual word embeddings from BERT. For example, one simple way is to use the output from the final layer as the vector representation for the input sequence (the blue concatenated squares in figure 3.5). However, it is argued in the original paper [33] that the best way to generate contextual language embeddings is to concatenate the outputs of the last four layers, which achieves the best performance for the downstream task.

Figure 3.6 illustrates how to creates contextual word embeddings for a single word “identified”. Noticeably, the value of embeddings for the same word will be different in various contexts (sentences). For example, for the same word “bank”, its embeddings will be distinct in the following two sentences: “willows lined the **bank** of the stream.” and “someone robs a **bank**.”. This feature of contextual word embeddings enables better representations for inputs compared to non-contextual word embeddings such as word2vec [27].

Chemical/cancer embeddings

BioBERT [13], a specialized BERT model pre-trained on biomedical text, is used to create the contextual word embeddings for the chemicals and cancers. Training on biomedical text can ensure the ability of the BioBERT on dealing with biological terms. The creation of embeddings follows the procedure below.

Let a chemical or cancer name be w_0. PubChem and CTD database [41] are used to create a synonym list $S = \{w_0, w_1, \ldots, w_n\}$ for w_0, and every name $w_i \in S$ is a synonym for w_o. Secondly, a list of articles$A = \{a_1, a_2, \ldots, a_n\}$ are pulled from PubMed as the contexts for S, where a_i is the corresponding context for $w_i \in S$. Once the context list A is obtained, BioBERT will take these contexts as the inputs and ouput the embeddings for all the elements in the synonym list. The final embedding for w_0 will be the average of embeddings of all the synonyms of w_0.

For example, for the chemical w_0 = "amifostine " (a chemical used in cancer chemotherapy), the corresponding synonym list S = { amifostine, Ethyol, Ethiofos }, and the context list $A = \{a_1, a_2, a_3\}$ is defined as the following:

context a_1: ...to determine the effect of ***amifostine*** on the safety and efficacy of induction chemotherapy with high-dose cisplatin and vinblastine followed by large-field thoracic irradiation to 60 gy in patients with stage iiia or iiib non-small-cell lung cancer ...(nsclc).

context a_2: ...Treatment with 75 or 150 mg/kg of **Ethyol** prevents radiation-induced learning and transitory memory dysfunction in young rats....

context a_3: ...The present study was performed to analyze the in vitro effectiveness of light-activated merocyanine 540 phototreatment (LAMP) and an aminothiol (**ethiofos**) as a marrow-purging regimen for small cell lung cancer (SCLC)...

Figure 3.7 shows another example of generating embeddings for the term "lung cancer". It retrieves the synonyms of "lung cancer" from CTD databases; the creation of chemical and cancer embeddings follows the same procedure.

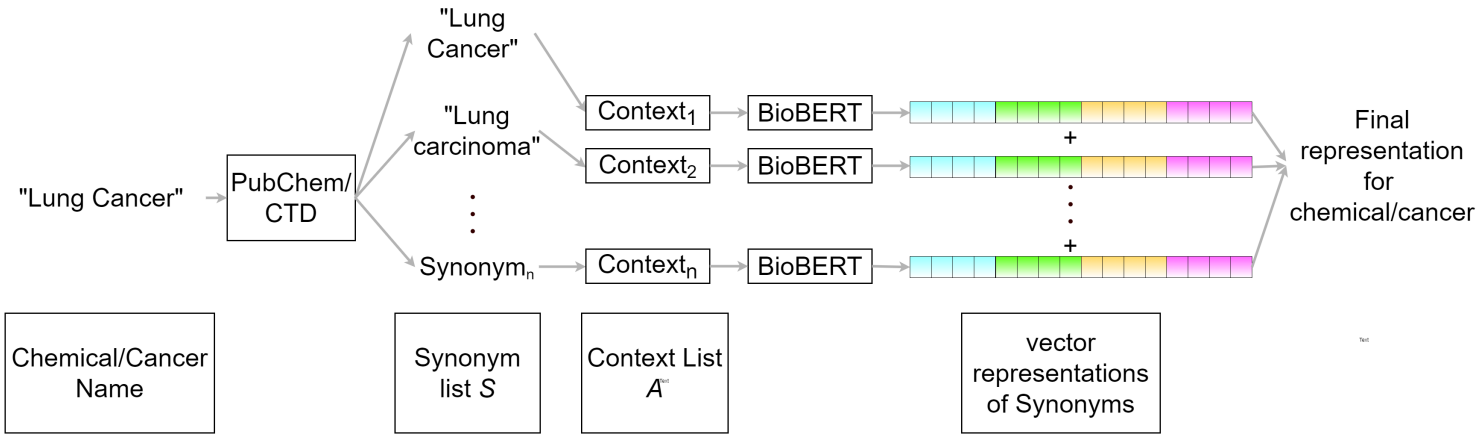

Figure 3.7: The process of embedding creation for chemicals/cancers. From left to right, the system takes the term "Lung Cancer" as the input. Firstly, a list of synonyms S for the term is created, and then every synonym is mapped to a context. BioBERT takes the context as the input, producing the embeddings for the synonym. The final representation for "lung cancer" will be the average of embeddings of synonyms.

These resulting embeddings for chemicals and cancers can then replace one-hot encodings in BIO-RGCN framework.

Out of vocabulary items

While this approach works well for cancers, the generation of chemical embeddings requires treatment on out of vocabulary items. BERT model maintains a list of vocabulary at the training time, and words which are not in the vocabulary list are classified as out of vocabulary items.

BERT model will continue to break down a out of vocabulary item into sub-words until all the sub-words are in the vocabulary list. As a result, when generating embeddings for the complex chemical names, these chemical names will be broken down to sub-words. For example, the chemical "Crizotinib" will be broke down to "c", "riz","ot","ini", and "b" by BERT model with associated vector embeddings. To get the embedding for the original chemical name, I take the vector representations of the sub-words and compute the average of these vectors as the final representation for chemicals.

The advantages of using NLP embeddings

With the approach mentioned above, I successfully create vector representations for all the chemicals and cancers in the CA-CHEM dataset. There are two advantages of replacing one-hot encoding with NLP embeddings for chemicals and cancers. Firstly, these vector representations can potentially improve the accuracy of chemical-cancer link prediction task because embeddings encode millions of biomedical articles.

More importantly, the second advantage is that using NLP embeddings enables the trained system to make a prediction on unseen chemicals and cancers. The link prediction systems with GNNs can be applied to nodes in the training set; for unseen nodes, it is required that the representations of unseen nodes have a similar distribution to the nodes in the training set. With the NLP embeddings, we can create vector representations for the unseen chemicals/cancers, and these vectors have the same distribution as the vector representations of existing chemical/cancer nodes. Therefore, BIO-

RGCN system is able to make a prediction for new chemical and cancer nodes.

Chapter 4

Evaluation for cancer-chemical association prediction

This chapter will present the evaluation result of BIO-RGCN framework for chemical-cancer association prediction. In addition, the contextual word embeddings for diseases and chemicals are evaluated through the dimension reduction and probing techniques.

4.1 Chemical-cancer link prediction

CA-CHEM dataset (from chapter-3) is used to evaluate the performance of BIO-RGCN on the chemical-cancer link prediction task. In order to obtain a more reliable conclusion, an ablation study is conducted by removing different components from BIO-RGCN model. An ablation analysis compares different versions of models on the same dataset.

Figure 4.1 shows different models that will be compared. The baseline model (BM) consists of chemical, cancer, and gene nodes with only one-hot encoding; the second model, Gene-emd model (GM), uses the GCN encoder to encode gene-gene networks and replace the one-hot encoding for genes; the third model, Gene-emd model with NLP (GMnlp), uses BioBERT to create embeddings for chemicals and cancers, and these embeddings are integrated

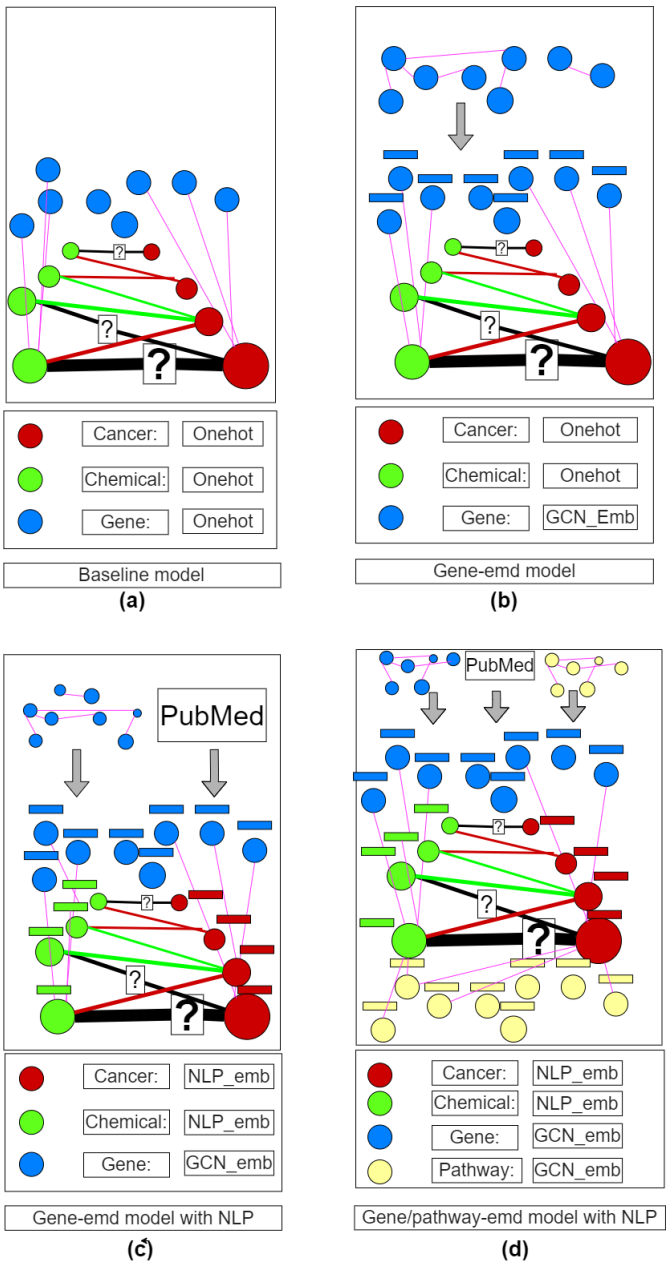

Figure 4.1: Different models for the ablation analysis. (GCN stands for graph convolutional network encoder while emd is the short for embeddings.) (a) Baseline model uses one-hot encoding to represent all the nodes. (b) Gene-emd model use trained embeddings to represent gene nodes. (c) Gene-emd model with NLP adds NLP embeddings for cancer and chemical nodes. (d) The last model integrate pathway-pathway information in the form of embeddings.

	Values
The number of RGCN layer	2
The number of hidden states of RGCN layer	32
Dropout rate	0.25
Optimization Algorithm	Adam
Epochs	500
Learning rate of the optimizer	0.01

Table 4.1: Hyper-parameter values for the ablation study

.

into the heterogeneous network. The last model, Gene/pathway-emd model with NLP (GPMnlp), adds pathway information to the network. All the hyper-parameters for these models will be the same; the details of these parameters are shown in table-4.1. The Adam [44] algorithm is used to optimize the cost function, and the dropout rate refers to the probability of an element to be zeroed.

Two numerical metrics are used for evaluation: **average precision (AP)** and **area under the receiver operating characteristic curve (AUC)**. AP is a single-value metric summarizing the precsion-recall curve, which is defined in equation-4.1:

$$AP = \sum_{1}^{n} (R_n - R_{n-1}) P_n \tag{4.1}$$

where R_n/P_n are the precision and recall at nth threshold.

I split the dataset into a training set, validation set, and testing set with a ratio of $8:1:1$, and four models are evaluated on the test set. The result is shown in table-4.2; I train every model for five times to account for the variability of the result. It is shown that GMnlp has the best performance in terms of both AP and AUC; AUC of GMnlp is 6% higher than the baseline model while the AP is 5% higher. The use of NLP embeddings and gene embeddings are clearly beneficial for the link prediction task because of the improved performance of GM and GMnlp compared to the baseline model. However, the use of pathway information decreases the performance of the

	BM	GM	GMnlp	GPMnlp
AUC	0.748 ± 0.001	0.797 ± 0.002	**0.819 ± 0.002**	0.806 ± 0.003
AP	0.729± 0.001	0.733 ± 0.003	**0.776 ± 0.004**	0.755± 0.001

Table 4.2: Average precision and AUC of fours models for the chemical-cancer link prediction task with standard errors.

.

	BM	GM	GMnlp	GPMnlp
BM		**0.0021**	**0.0014**	**0.0016**
GM			**0.0039**	**0.0040**
GMnlp				0.059

Table 4.3: The p-values of Statistical significance tests for AUC

.

GPMnlp. The reason could be that pathway information overlaps with gene information, and the use of pathways embeddings bring noises to the training process, resulting in a decrease in the model performance. Another observation from the table is that the values of AP and AUC correlate with each other positively across different models, indicating the stability of models to deal with positive and negative examples.

Statistical significance testing

Furthermore, I conduct statistical significance tests to compare four models[1]. Table-4.3 shows the result of statistical tests on AP. As the test results indicate, all the systems with external embeddings is better than the baseline system. There is also a difference between the performance of GMnlp andGM, . The performance of GMnlp and GPMnlp is very similar to each other, which is consistent with AP and AUC metrics.

In summary, the ablation study shows that BIO-RGCN is an effective framework for integrating the information on the heterogeneous network. The gene embeddings from gene-gene network improve the performance compared to the baseline model. The embeddings for chemicals and cancers from

[1]Permutation test is used here. The significance level equals 0.05: if the p-value of a sign test between two systems is smaller than this threshold, we would say there are significant differences between the two systems.

BioBERT has a positive influence on the performance of the system (AP and AUC improve) while pathway information does not improve the performance. Moreover, NLP embeddings for chemicals and cancers are important for the system since it enables the system to work on unseen chemicals and diseases. Finally, due to the overlap between gene information and pathway information, the inclusion of pathway information does not help with the prediction task.

4.2 Evaluation of the NLP embeddings

In this section, the NLP embeddings for cancers and chemicals will be evaluated through the dimension reduction algorithm and probing. The embeddings for cancer and chemicals are created through BioBERT, denoted as h_{ca} and h_{chem}, the dimension of these embeddings are 3072.

Exploration of embeddings space with t-SNE

I first apply t-SNE [48] to the high-dimensional embeddings to visualize the distribution of different entities. t-Distributed Stochastic Neighbor Embedding (t-SNE) is a technique to visualize high-dimensional data. t-SNE can not only reveal the structure of the data dynamically from different perspectives, but it avoids the crowding of data points to provide better visualization. t-SNE uses Student-t distribution to compute the similarity score between two points and minimizes the Kullback-Leibler divergence between high-dimensional representations and low-dimensional representations of the data points.

Figure 4.3 shows the result of t-SNE applied to chemical and cancer embeddings. One pattern persists through the training process of t-SNE is that the cancer representations tend to cluster, and there is a clear boundary between chemicals and cancers. This pattern indicates that NLP embeddings have the ability to differentiate two types of entities. Furthermore, a closer look at the plot reveals that the chemicals which have properties tend to cluster together, as shown in figure 4.2. The clustering of different chemicals of the

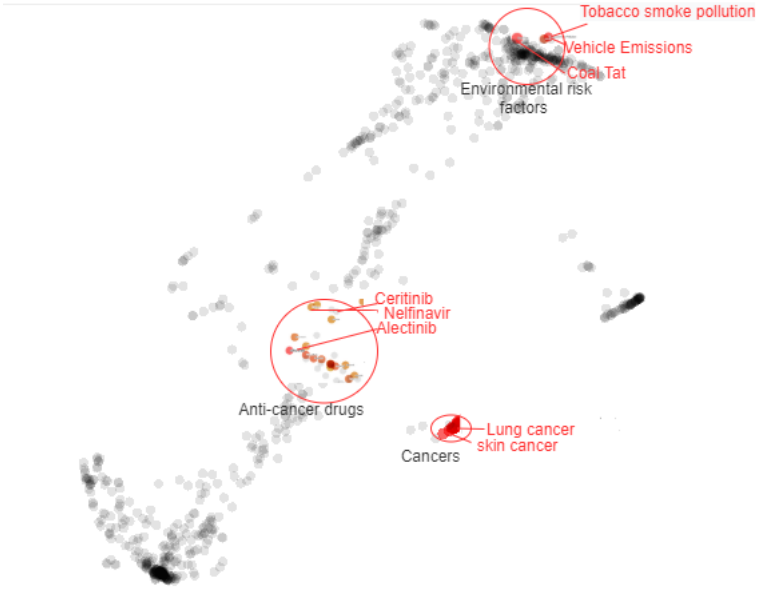

Figure 4.2: t-SNE visualization of my corpus with annotations. The chemicals that share common properties tend to cluster in the diagram. Ceritinib, Nelfinavir, and Alectinib are all anti-cancer drugs.

same properties reinforcement the assumption, the NLP embeddings from BioBERT can capture the semantic meaning of chemicals and cancers from biomedical text.

Classifier probing for understanding embeddings

Probing techniques are used to quantify the ability of NLP embeddings to capture the semantic meaning of chemicals and cancers. Probing refers to a class of techniques to explain the black-box machine learning models or trained embeddings. In recent years, it has been used to explore the pre-trained language models in NLP [49, 50, 51]. The idea of probing is the following: high-dimensional embeddings or complex black-box machine learning models often lacks interpretability. In order to understand what is entailed in models and embeddings, people directly apply these models and embeddings to some simple tasks and observe the performance without much further training; performance on the simple tasks would indicate the internal information encoded in the model or embeddings.

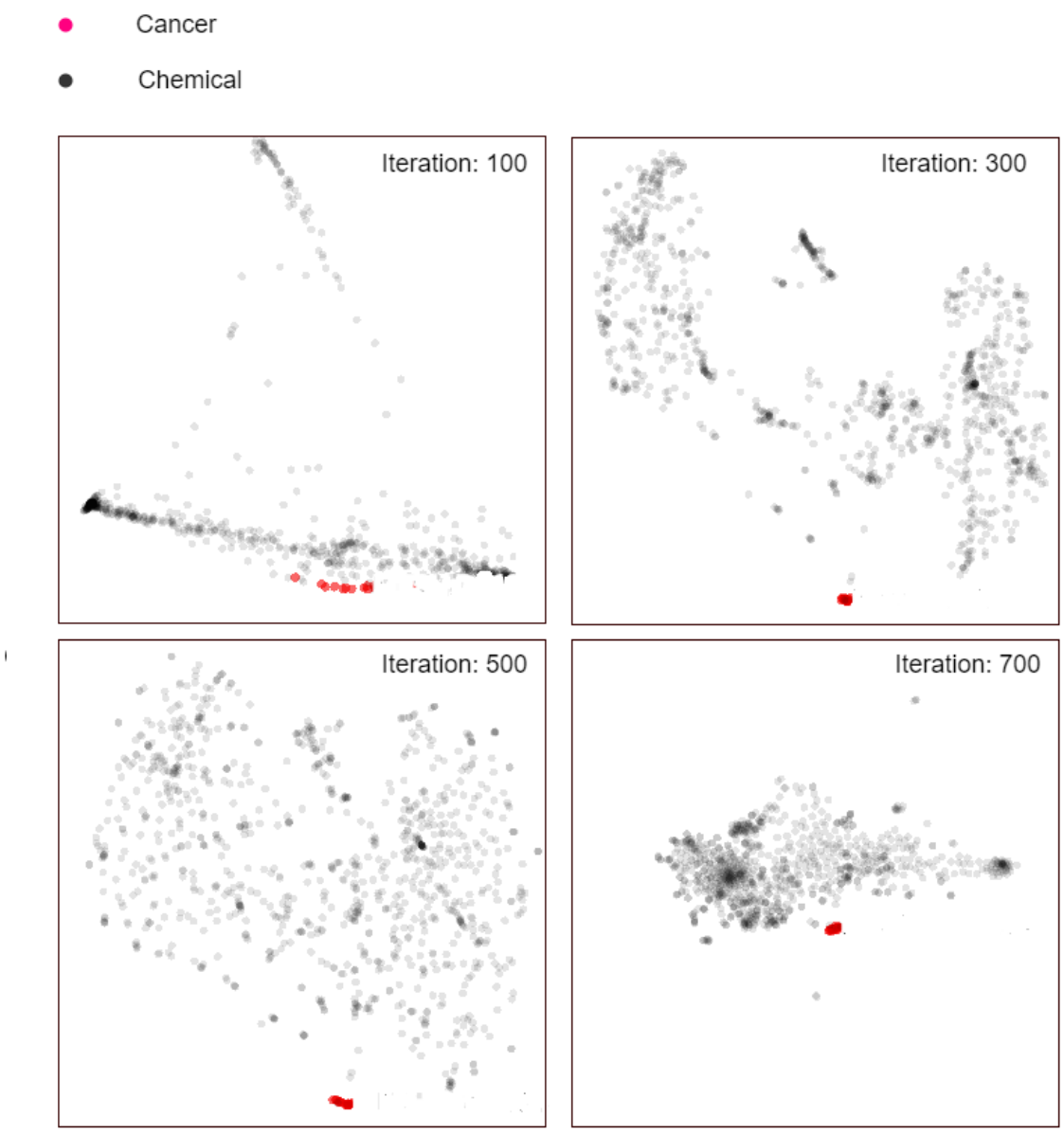

Figure 4.3: The t-SNE result of embeddings for chemicals and cancers. The iteration number indicates when the plot is made during the training process of t-SNE. At the beginning, the data points spread out, and they get closer and closer during the training process. Throughout the process, there is a clear boundary between cancers and chemicals.

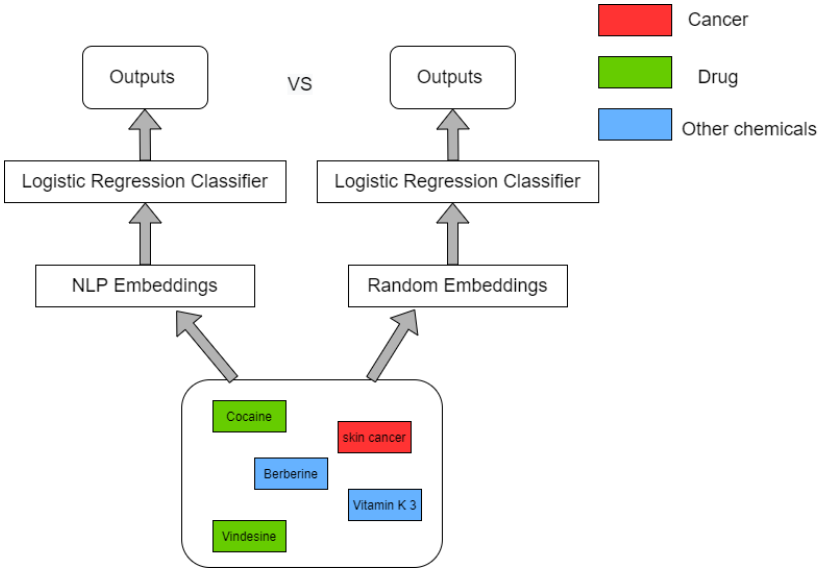

Figure 4.4: The workflow of classification probing to reveal the NLP embeddings. The ouputs from two approches are compared against each other.

In this work, I use one type of probing techniques called classifier probing to crystalize the internal information encoded in NLP embeddings of chemicals and cancers. I first design an entity classification tasks: 205 entities including cancers, chemicals and drugs are selected and annotated manually with corresponding labels; there are three types of entities: "C" represents Cancer, "D" represents Drug, and "OC" represents other chemicals. The task is to classify the type of entities given the corresponding entity embeddings.

In order to evaluate the NLP embeddings from BioBERT, a simple logistic regression classifier is trained to classify the types of entities given the NLP embeddings from BERT while a baseline method uses randomly generated embeddings and a logistic regression classifier. The dataset containing 205 entities is split into training and test set with a ratio of $8:2$. The validation set is not required because the default settings for logistic regression is used. Better performance of the model with NLP embeddings will indicate that its ability to encode biomedical information. Figure 4.4 provides a graphical illustration for classifier probing.

It turns out there is a significant difference between the performance of the

	NLP embeddings	Baseline
Mean Accuracy	**0.77** $\pm$ 0.01	0.45 $\pm$ 0.04

Table 4.4: Classifier probing. The model with NLP embeddings is better than the baseline model with randomly generated embeddings by a large margin.

baseline model and the model with NLP embeddings. Table-4.4 shows the performance of two models. NLP embeddings perform much better than the baseline model measure by the mean accuracy for the multi-class classification task. It indicates that NLP embeddings from BioBERT encode different entities (drugs, chemicals, environmental risk factors) in a reasonable way based on the evidence from millions of biomedical articles. Furthermore, these embeddings potentially encode more fine-grained categories and even dependencies between individual entities.

Chapter 5

Different applications of model

In this chapter, I will demonstrate the generalization ability of BIO-RGCN system for other link prediction tasks. In particular, as a case study, BIO-RGCN is applied to drug-target interactions (DTIs) prediction, which is important for fighting different types of diseases.

5.1 Task specification

Drugs become effective by interacting with certain protein targets in the human body. An essential step in discovering new drugs is to predict the target proteins of drugs, and this step is referred to as **DTIs prediction**. Traditional methods for DTIs prediction are often time-consuming and require the use of 3-d structures of proteins and drugs; with the development of computational pharmacology, many data-driven approaches have been proposed for DTIs prediction [45, 46, 47]. Despite the difference between DTIs and chemical-cancer association prediction, both of them can be formulated as a link prediction task on heterogeneous networks (fig-5.1).

As figure 5.1 shows, this chapter will analyze one type of network that includes disease nodes as extra information. The task is to predict the linkage between drug and protein nodes with protein-protein, drug-drug, disease-drug, and disease-protein interactions. It is clear that the structure of this

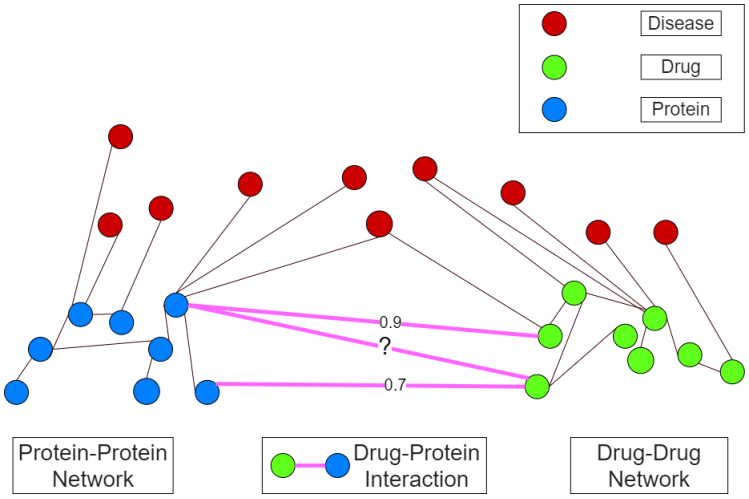

Figure 5.1: DTIs prediction. There are three types of node in the network. The task is to predict the linkage between protein and drug nodes given other information (protein-protein, drug-drug, and disease nodes).

task is similar to the task of predicting chemical-cancer association. Therefore, BIO-RGCN can be applied to solve this problem.

5.2 Implementation details

Given a task to predict the association between type A nodes and type B nodes in heterogeneous networks, BIO-RGCN can be used through the following steps:

1. Decompose the heterogeneous graph into two parts: bipartite graph G_{bipart} and additional graph $G_{additional}$.
2. Use GCNs to generate embeddings for the nodes in the $G_{additional}$.
3. Use R-GCN to generate embeddings for of type A, B nodes in G_{bipart}.
4. Define a decoder and cost function with the embeddings of A, B nodes.

As figure 5.2 shows, when using BIO-RGCN to predict DTIs, the additional network $G_{addtional}$ includes protein-protein network and drug-drug network while G_{bipart} includes all the components of the heterogeneous network except

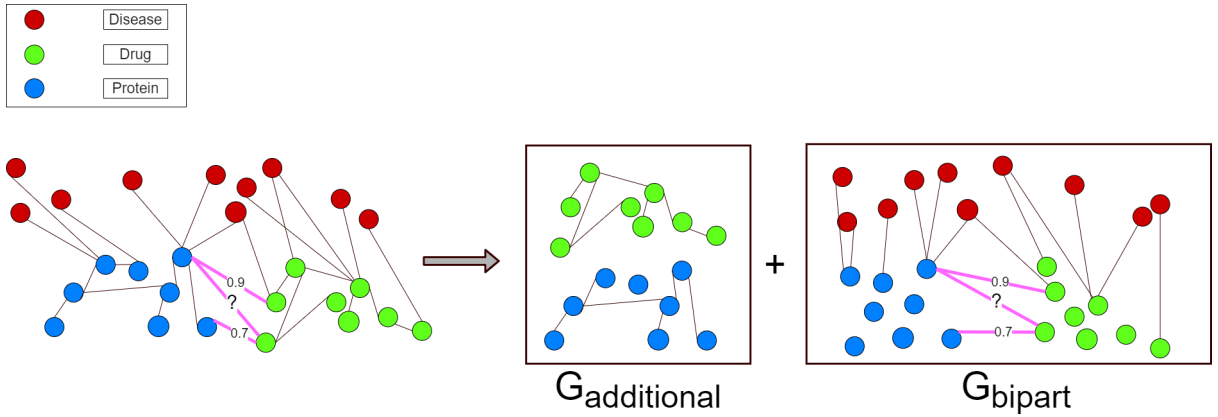

Figure 5.2: Decomposition of DTIs networks. The heterogeneous network is decomposed into $G_{additional}$ and G_{bipart}; $G_{additional}$ includes protein-protein and drug-drug interactions while G_{bipart} includes all the remaining parts.

	BIO-RGCN	DTINet	BIO-RGCN-pre-trained
AUC	0.9351 ± 0.007	0.9137 ± 0.001	**0.9401 ± 0.002**
AP	0.9141 ± 0.002	0.9319 ± 0.003	**0.9395 ± 0.003**

Table 5.1: Average precision and AUC of BIO-RGCN and DTINet for the DTIs prediction task with standard errors.

.

for the protein-protein edges and drug-drug edges. Following the procedure, first, I use $G_{additional}$ to create embeddings for drug and proteins with GCN encoder. Secondly, these embeddings are used to initialize drug, protein nodes in G_{bipart} while disease nodes are initialized with one-hot encoding. G_{bipart} is then encoded with R-GCN encoder to create further embeddings for drug and diseases (the details of GCN and R-GCN encoders can found in chapter-3)

Assuming the resulting embeddings for drug i and disease j are $z_i \in \mathbb{R}^m$ and $z_j \in \mathbb{R}^m$, the decoder and cost function can be defined in equation-5.1, 5.2.

$$\text{Decoder: } p(i,j) = sigmoid(z_i^T M z_j) \tag{5.1}$$

where $p(i,j)$ represents the probability of edge (i,j) exists. $M \in \mathbb{R}^m \times m$ is a weight matrix encoding the interaction between two embeddings.

$$\text{Cost function: } C = \sum_{(i,j)\in E_{dp}} -\log(p(i,j)) - \sum_{(m,n)\in E_{dp}^{\complement}} \log(1-p(m,n)) \tag{5.2}$$

where E_{dp} represents the edge set containing all the interactions between drugs and proteins while $E_{dp}^{\complement}$ represents the negatively sampled edge set. Minimizing the cost function C results in the optimal parameters for the encoders and the decoder. With equation-5.1, the probability for the existence of a drug-protein edge can be computed.

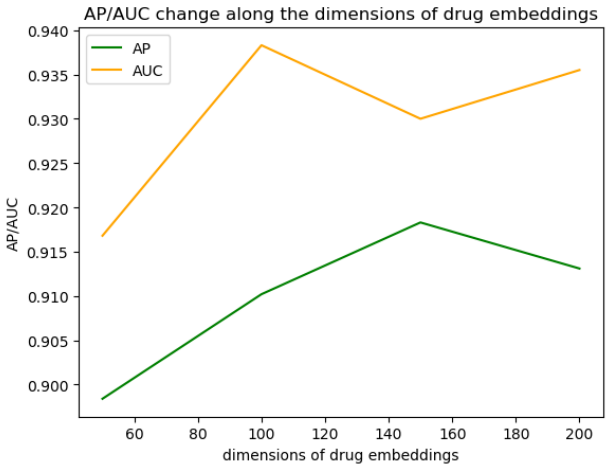

Figure 5.3: The change of AP/AUC on DTIs prediciton with the dimensions of drug embeddings.

5.3 Evaluation for DTIs prediction

This section will evaluate the performance of BIO-RGCN framework on the DTIs prediction task. The dataset used for evaluation is created by Luo et al. [10], in which they propose a machine learning framework - DTINet - to prediction DTIs. I will compare the performance of BIORGCN with the DTINet, which enables the model to be indirectly compared with BLMNII, NetLapRLS, and HNM [45, 46, 47] since DTIs performs better than these systems on the same dataset.

The structure of the dataset has been shown in figure 5.1. There are 708 drug, 1,512 proteins, 5,603 diseases, and 1,923 drug-protein interactions in the heterogeneous network. In addition to the network data, the dataset also contains pre-trained vectors for drugs and proteins which incorporate the 3-d drug-drug similarity and protein-protein similarity information. BIO-RGCN is applied to predict drug-protein interactions with the same hyper-parameter settings as 4.1. Table-5.1 summaries AP and AUC of various models. BIO-RGCN-pre-trained is the model using pre-trained vectors for drug and protein resulting from 3-d similarity scores. While BIO-RGCN has a lower AP compared to DTINet, BIO-RGCN-pre-trained performs better than

DTINet, it indicates that the use of 3-d structure information can improve the performance of the system.

To test the robustness of the system, I run several experiments with different size of drug vectors. The result is presented in figure 5.3. Across different dimensions of the drug embeddings, the AP is always above 0.9, and AUC is above 0.912, which proves the stability of BIO-RGCN model.

5.4 Principals of using BIO-RGCN

In this section, I will provide several principals for generalizing BIO-RGCN framework to link prediction tasks on heterogeneous networks. The content will be arranged in the form of questions and answers.

What types of task can BIO-RGCN be applied to?

In theory, BIO-RGCN can be applied to different link prediction tasks on heterogeneous networks while this project focuses on predicting the links between **two** types of nodes. Suppose the task is to predict the links between m types of nodes $V = \{v_1, v_2, \ldots, v_m\}$ on G. BIO-RGCN decomposes the given networks G into $G_{additonal}$ and G_{bipart}. $G_{additional}$ includes the sub-networks except for target edges; G_{bipart} should include all the edges that the user aims to predict and edges that connect the nodes in V and other types of nodes.

While the framework can be extended to a massive network which includes different types of nodes, the performance of the system is expected to drop since the **indirect links** cannot propagate the information to target nodes efficiently. As shown in figure 5.4, with the same number of nodes in two networks and the same prediction task, BIO-RGCN would be a more appropriate tool for the second type of networks.

How to add side information as additional nodes and embeddings? Any additional information which potentially relates to the link prediction task can be included in forms of addition nodes or additional embeddings. For example, if someone wants to include the microRNA(miRNA) information

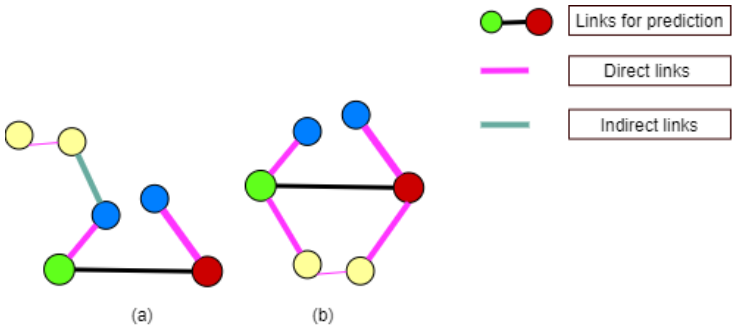

Figure 5.4: The types of graph BIO-RGC can work with. Yellow and blue nodes are additional nodes; the task is to predict the link between green and red nodes. In (a), there is an indirect link between yellow nodes and the target node (green node), which could introduce noises when information goes from yellow nodes to the target node. In (b), all the additional nodes are directly connected to target nodes. BIO-RGCN would be more applicable to networks like (b).

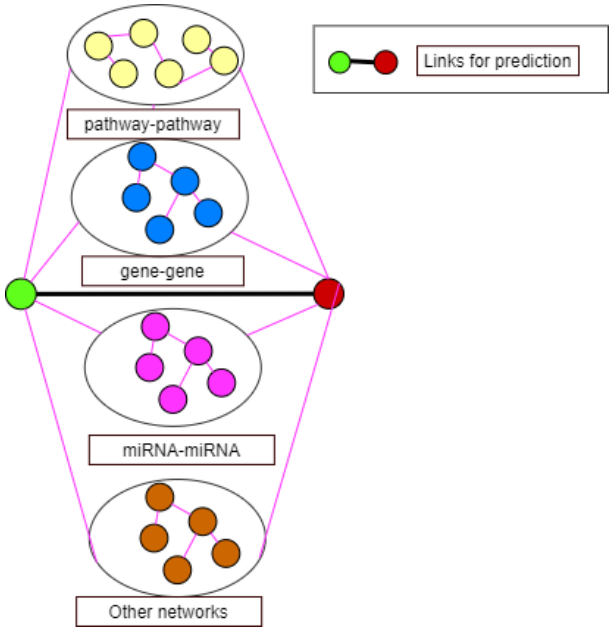

Figure 5.5: External information added in the form of extra nodes. Any other information can be added to the heterogeneous network in the same way as gene-gene networks and pathway-pathway networks.

for the chemical-cancer association task, the miRNA network can be added as shown in figure 5.5. The miRNA network will be included in $G_{additonal}$, and embeddings for miRNA nodes can be created through the GCN encoder. The framework also enables the addition of an arbitrary number of other networks.

BIO-RGCN also allows extra information to be added as the external embeddings, and the embeddings can be added in the same way as NLP embeddings for cancers and chemicals in the CA-CHEM dataset. Any type of nodes in the network can be initialized with known embeddings to help with the link prediction process. These embeddings can potentially encode information that cannot be expressed in a graphical way, and thus they are essential for many link prediction tasks. However, when there is no external embeddings for the node, the node will be initialized with one-hot encoding.

Chapter 6

Demonstration of model

This short chapter will present some innovative chemical-cancer links from BIO-RGCN model. Innovative links are defined as links that do not appear in the training set. A tool is created to demonstrate the prediction result[1]. This tool enables the user to look up the chemical-cancer associations, and it can also visualize NLP embeddings of chemicals and cancers with different dimension reduction algorithms. There are two types of chemical-cancer links.

- Treatment: The chemical has a potential therapeutic effect on cancer.
- Risk factor: Exposure to the chemical could lead to cancer, or the accumulation certain of the chemical correlates with cancer.

For any given pair of nodes $(chemical, cancer)$, BIO-RGCN model is able to assign two scores p_t, p_r to indicate the type of links between two nodes, where p_t is the probability of the link type being treatment while p_r represents the probability being a risk factor.

[1]The tool can be accessed through the following link; it links to a Colab Jupyter notebook. Open the link and then make a local copy, which would enable the use of this tool. `https://colab.research.google.com/drive/18ZTZYMXKOGT-xtpKHir11QznWhfWomZG?usp=sharing`

Names	Verified	Types
Doxorubicin	Yes	chemotherapy medication
Sorafenib	Yes	other medication
Famotidine	No	other medication
Quercetin	Yes	plant extracts
Orlistat	Yes	other medication
doxifluridine	Yes	chemotherapy medication
Oxaliplatin	Yes	chemotherapy medication
Leucovorin	Yes	chemotherapy medication
Capecitabine	Yes	chemotherapy medication
Celecoxib	Yes	other medication

Table 6.1: Top-10 chemicals of treatment effects for non-small-cell lung cancer from model predictions.

Names	Verifiable	Types
Tretinoin	Yes	other medication
Curcumin	Yes	plant extract
Cimetidine	Yes	other medication
Carboplatin	Yes	chemotherapy medication
octapeptide	Yes	other medication
fosbretabulin	Yes	other medication
Epirubicin	Yes	chemotherapy medication
Docetaxel	Yes	chemotherapy medication
Troglitazone	Yes	other medication
Pemetrexed	Yes	chemotherapy medication

Table 6.2: Top-10 chemicals of treatment effects for colorectal cancer from model predictions.

6.1 Chemicals with treatment effects

Table-6.1 shows top-10 chemicals which have treatment effects to non-small-cell lung cancer as predicted by the model, and table-6.2 shows the same statistics for colorectal neoplasms. These chemicals are classified into three categories: chemotherapy medications, plant extracts, and other medications. Chemotherapy medications are a type of drugs used for restricting the reproduction of cancer cells, while plant extracts are substances from natural plants. Drugs other than chemotherapy medications are classified as other

medications. I check the validity of predicted chemicals, and most of the chemicals can be verified with existing evidence. For example, Sorafenib is a drug developed for treating kidney cancer and liver cancer, and BIO-RGCN model predicts that Sorafenib can also be used for non-small-cell lung cancer (NSCLC). Many studies have shown that the use of Sorafenib indeed benefits a subset of NSCLC patients with a specific type of mutations [52, 53, 54, 55]. Similarly, another type of predicted chemical - retinoids - are proven to be helpful for controlling the colorectal cancer cell proliferation according to recent studies [56, 57, 58]. In general, among the top 10 treatment chemicals for 15 types of cancers, more than 95% of association can be verified with existing medical literature.

6.2 Chemicals as risk factors

Instead of analyzing risk factors for specific types of cancer, a ranked list of risk factors across different types of cancer will be analyzed. These results from BIO-RGCN are compared with existing lists of cancer causes from International Agency for Research on Cancer (IARC). IARC classifies more than $1,000$ likely cause for cancers into the following categories:

- Group 1: Carcinogenic to humans
- Group 2A: Probably carcinogenic to humans
- Group 2B: Possibly carcinogenic to humans
- Group 3: Not classifiable as to its carcinogenicity in humans

The top 15 predicted risk factors across different types of cancers are shown in table-6.3. Among 15 predicted risk factors, four of them are proven to be carcinogenic to humans, and seven of them are classified as 2A group. For other chemicals which are labelled as unknown, three of them, PhIP, Chloroprene, and N-Nitroso-N-methylurea are labelled "Reasonably anticipated to be human carcinogens" by National Toxicology Program (NTP) [59]. Finally, two types of chemicals, DBA and Fonofos do not appear in the investigation list of NTP and IARC, but existing medical literature has proven that they

Names	IARC class
Polychlorinated Biphenyls	Group 1
PhIP	Unknown*
DBA	Unknown
N-Nitrosodimethylamine	Group 2A
Vinyl Chloride	Group 2A
Arsenic	Group 1
N-nitrosodiethylamine	Group 2A
Benzo(a)pyrene	Group 1
Chloroprene	Unknown*
1,2,3-trichloropropane	Group 2A
diisopropanolnitrosamine	Unknown
N-Nitroso-N-methylurea	Unknown*
4-biphenylamine	Group 1
N-Nitrosodiethylamine	Group 2A
2-amino-3-methylimidazo(4,5-f)quinoline	Group 2A
1,2-Dimethylhydrazine	Group 2A
Fonofos	Unknown

Table 6.3: Top-15 chemicals as risk factors for 15 types of cancers from BIO-RGCN.

are potential candidates for carcinogenic substance for humans [60, 61].

In summary, the predicted chemicals from BIO-RGCN are valuable in the sense that the prediction is consistent with existing medical experiments. The model is able to recommend reasonable candidates to provide insights into drug repurposing, and it is also able to identify critical risk factors for cancers.

Chapter 7

Conclusions

7.1 Contributions

This project successfully develops a framework (BIO-RGCN) to predict the associations between chemicals and cancers with graph neural networks. Furthermore, natural language processing embeddings enable the framework to handle unseen inputs. In addition to chemical-cancer prediction task, the framework is applied to predict drug-target interactions, showing its abilities to be generalized to different types of biological networks. The performance of the framework is evaluated on DTINet dataset and the customized dataset (CA-CHEM). The numerical results confirm the stability of the framework on two tasks; qualitative analysis on predicted chemicals-cancer links are consistent with existing medical literature.

Another achievement of this project is that it proves the effectiveness of NLP embeddings as a way to extract information from millions of biomedical papers. The extracted embeddings can not only encode the syntactic structure of different entities but valuable properties such as the type of entities. Although this project only creates embeddings for cancers and chemicals, the same procedure can be applied to other types of biological entities such as proteins, genes, and other diseases. These embeddings can then be applied to downstream tasks to improve the performance.

7.2 Future work

Due to the interdisciplinary nature, this project can be explored in many directions in the future.

- Interpretability: Interpretability of machine learning models is essential for their applications in medical science. Graph neural networks (GNNs) are different from other types of neural networks, and many frameworks have been developed to interpret GNNs. For example, GNN-Explainer provides interpretation for link prediction and node classification results on GNNs by relating the predictions to neighbourhood nodes [62]. XGNN provides an explanation to GNNs at model level using reinforcement learning [63]. For this work, BIO-RGCN can be combined with frameworks like GNN-Explainer to increase the interpretability of predicted results.

- MicroRNA: In this work, I use pathway interactions and genes interactions information as the additional information to predict chemical-cancer associations; however, it is shown that microRNA also plays an important role in regulating various diseases including cancers [64, 65, 66, 67]. An Inclusion of microRNA could potential benefits the prediction of chemical-cancer associations.

- Experiments: Although the predictions from BIO-RGCN have been agreed upon by existing literature, further medical experiments are needed to confirm unprecedented predictions. Clinical trials and laboratory experiments are often used to confirm chemical-cancer associations. For drug-target interactions, ligand-based and docking approaches can be applied to verify drug protein associations [68, 69].

Bibliography

[1] Cancerresearchuk.org. 2020. [online] Available at: https://www.cancerresearchuk.org/sites/default/files/state_of_the_nation_april_2019.pdf Accessed 28 May 2020.

[2] Dewar, S.L. and Porter, J., 2018. The Effect of Evidence-Based Nutrition Clinical Care Pathways on Nutrition Outcomes in Adult Patients Receiving Non-Surgical Cancer Treatment: AS ystematic Review. Nutrition and cancer, 70(3), pp.404-412.

[3] Ou, S.H.I., Bartlett, C.H., Mino-Kenudson, M., Cui, J. and Iafrate, A.J., 2012. Crizotinib for the treatment of ALK-rearranged non-small cell lung cancer: a success story to usher in the second decade of molecular targeted therapy in oncology. The oncologist, 17(11), p.1351.

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

[5] Schlichtkrull, M., Kipf, T.N., Bloem, P., Van Den Berg, R., Titov, I. and Welling, M., 2018, June. Modeling relational data with graph convolutional networks. In European Semantic Web Conference (pp. 593-607). Springer, Cham.

[6] Donald, B.R., 2011. Algorithms in structural molecular biology. MIT Press.

[7] Veselkov, K., Gonzalez, G., Aljifri, S., Galea, D., Mirnezami, R., Youssef, J., Bronstein, M. and Laponogov, I., 2019. HyperFoods: Machine intelligent mapping of cancer-beating molecules in foods. Scientific reports, 9(1), pp.1-12.

[8] Zitnik, M., Agrawal, M. and Leskovec, J., 2018. Modeling polypharmacy side effects with graph convolutional networks. Bioinformatics, 34(13), pp.i457-i466.

[9] Whitebread, S., Hamon, J., Bojanic, D. Urban, L. Keynote review: in vitro safety pharmacology profiling: an essential tool for successful drug development. Drug Discov. Today 10, 1421–1433 (2005).

[10] Luo, Y., Zhao, X., Zhou, J., Yang, J., Zhang, Y., Kuang, W., Peng, J., Chen, L. and Zeng, J., 2017. A network integration approach for drug-target interaction prediction and computational drug repositioning from heterogeneous information. Nature communications, 8(1), pp.1-13.

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

[12] Zitnik, M., Nguyen, F., Wang, B., Leskovec, J., Goldenberg, A. and Hoffman, M.M., 2019. Machine learning for integrating data in biology and medicine: Principles, practice, and opportunities. Information Fusion, 50, pp.71-91.

[13] Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C.H. and Kang, J., 2020. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. Bioinformatics, 36(4), pp.1234-1240.

[14] Gondane, S., 2019, August. Neural Network to identify personal health experience mention in tweets using BioBERT embeddings. In Proceedings of the Fourth Social Media Mining for Health Applications (SMM4H) Workshop Shared Task (pp. 110-113).

[15] Telukuntla, S.K., Kapri, A. and Zadrozny, W., 2019, September. UNCC biomedical semantic question answering systems. BioASQ: Task-7B, Phase-B. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases (pp. 695-710). Springer, Cham.

[16] Beltagy, I., Cohan, A. and Lo, K., 2019. Scibert: Pretrained contextualized embeddings for scientific text. arXiv preprint arXiv:1903.10676.

[17] Wei, C.H., Allot, A., Leaman, R. and Lu, Z., 2019. PubTator central: automated concept annotation for biomedical full text articles. Nucleic acids research, 47(W1), pp.W587-W593.

[18] Wei, C.H., Kao, H.Y. and Lu, Z., 2013. PubTator: a web-based text mining tool for assisting biocuration. Nucleic acids research, 41(W1), pp.W518-W522.

[19] Pyysalo, S., Baker, S., Ali, I., Haselwimmer, S., Shah, T., Young, A., Guo, Y., Högberg, J., Stenius, U., Narita, M. and Korhonen, A., 2019. LION LBD: a literature-based discovery system for cancer biology. Bioinformatics, 35(9), pp.1553-1561.

[20] Krizhevsky, A., Sutskever, I. and Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).

[21] Gilmer, J., Schoenholz, S.S., Riley, P.F., Vinyals, O. and Dahl, G.E., 2017, August. Neural message passing for quantum chemistry. In Proceedings of the 34th International Conference on Machine Learning-Volume 70 (pp. 1263-1272). JMLR. org.

[22] Yosinski, J., Clune, J., Bengio, Y. and Lipson, H., 2014. How transferable are features in deep neural networks?. In Advances in neural information processing systems (pp. 3320-3328).

[23] Do, C.B. and Ng, A.Y., 2006. Transfer learning for text classification. In Advances in Neural Information Processing Systems (pp. 299-306).

[24] Radford, A., Narasimhan, K., Salimans, T. and Sutskever, I., 2018. Improving language understanding by generative pre-training. URL https://s3-us-west-2. amazonaws. com/openai-assets/researchcovers/languageunsupervised/language understanding paper. pdf.

[25] Lample, G. and Conneau, A., 2019. Cross-lingual language model pretraining. arXiv preprint arXiv:1901.07291.

[26] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K. and Fei-Fei, L., 2009, June. Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition (pp. 248-255). Ieee.

[27] Le, Q. and Mikolov, T., 2014, January. Distributed representations of sentences and documents. In International conference on machine learning (pp. 1188-1196).

[28] Pennington, J., Socher, R. and Manning, C.D., 2014, October. Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP) (pp. 1532-1543).

[29] Bojanowski, P., Grave, E., Joulin, A. and Mikolov, T., 2017. Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics, 5, pp.135-146.

[30] Melamud, O., Goldberger, J. and Dagan, I., 2016, August. context2vec: Learning generic context embedding with bidirectional lstm. In Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning (pp. 51-61)

[31] Peters, M.E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K. and Zettlemoyer, L., 2018. Deep contextualized word representations. arXiv preprint arXiv:1802.05365.

[32] Howard, J. and Ruder, S., 2018. Universal language model fine-tuning for text classification. arXiv preprint arXiv:1801.06146.

[33] Devlin, J., Chang, M.W., Lee, K. and Toutanova, K., 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

[34] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł. and Polosukhin, I., 2017. Attention is all you need. In Advances in neural information processing systems (pp. 5998-6008).

[35] Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R.R. and Le, Q.V., 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In Advances in neural information processing systems (pp. 5754-5764).

[36] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L. and Stoyanov, V., 2019. Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692.

[37] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D. and Sutskever, I., 2019. Language models are unsupervised multitask learners. OpenAI Blog, 1(8), p.9.

[38] Wang, X. and Zhang, Z., 2020. Thunlp/Plmpapers. [online] GitHub. Available at: ¡https://github.com/thunlp/PLMpapers¿ [Accessed 30 May 2020].

[39] Gori, M., Monfardini, G. and Scarselli, F., 2005, July. A new model for learning in graph domains. In Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005. (Vol. 2, pp. 729-734). IEEE.

[40] Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P. and Bengio, Y., 2017. Graph attention networks. arXiv preprint arXiv:1710.10903.

[41] Davis, A.P., Grondin, C.J., Johnson, R.J., Sciaky, D., King, B.L., McMorran, R., Wiegers, J., Wiegers, T.C. and Mattingly, C.J., 2017. The comparative toxicogenomics database: update 2017. Nucleic acids research, 45(D1), pp.D972-D978.

[42] Stark, C., Breitkreutz, B.J., Reguly, T., Boucher, L., Breitkreutz, A. and Tyers, M., 2006. BioGRID: a general repository for interaction datasets. Nucleic acids research, 34(suppl_1), pp.D535-D539.

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

[44] Kingma, D.P. and Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

[45] Mei, J.P., Kwoh, C.K., Yang, P., Li, X.L. and Zheng, J., 2013. Drug–target interaction prediction by learning from local information and neighbors. Bioinformatics, 29(2), pp.238-245.

[46] Xia, Z., Wu, L.Y., Zhou, X. and Wong, S.T., 2010, September. Semi-supervised drug-protein interaction prediction from heterogeneous biological spaces. In BMC systems biology (Vol. 4, No. 2, p. S6). BioMed Central.

[47] Wang, W., Yang, S., Zhang, X. and Li, J., 2014. Drug repositioning by integrating target information through a heterogeneous network model. Bioinformatics, 30(20), pp.2923-2930.

[48] Maaten, L.V.D. and Hinton, G., 2008. Visualizing data using t-SNE. Journal of machine learning research, 9(Nov), pp.2579-2605.

[49] Linzen, T., Dupoux, E. and Goldberg, Y., 2016. Assessing the ability of LSTMs to learn syntax-sensitive dependencies. Transactions of the Association for Computational Linguistics, 4, pp.521-535.

[50] Tenney, I., Das, D. and Pavlick, E., 2019. Bert rediscovers the classical nlp pipeline. arXiv preprint arXiv:1905.05950.

[51] Liu, N.F., Gardner, M., Belinkov, Y., Peters, M. and Smith, N.A., 2019. Linguistic knowledge and transferability of contextual representations. arXiv preprint arXiv:1903.08855.

[52] Smit, E.F., Dingemans, A.M.C., Thunnissen, F.B., Hochstenbach, M.M., van Suylen, R.J. and Postmus, P.E., 2010. Sorafenib in patients with advanced non-small cell lung cancer that harbor K-ras mutations: a brief report. Journal of thoracic oncology, 5(5), pp.719-720.

[53] Kelly, R.J., Rajan, A., Force, J., Lopez-Chavez, A., Keen, C., Cao, L., Yu, Y., Choyke, P., Turkbey, B., Raffeld, M. and Xi, L., 2011. Evaluation of KRAS mutations, angiogenic biomarkers, and DCE-MRI in patients with advanced non–small-cell lung cancer receiving sorafenib. Clinical cancer research, 17(5), pp.1190-1199.

[54] Kim, E.S., Herbst, R.S., Wistuba, I.I., Lee, J.J., Blumenschein, G.R., Tsao, A., Stewart, D.J., Hicks, M.E., Erasmus, J., Gupta, S. and Alden, C.M., 2011. The BATTLE trial: personalizing therapy for lung cancer. Cancer discovery, 1(1), pp.44-53.

[55] Lind, J.S., Dingemans, A.M.C., Groen, H.J., Thunnissen, F.B., Bekers, O., Heideman, D.A., Honeywell, R.J., Giovannetti, E., Peters, G.J., Postmus, P.E. and van Suylen, R.J., 2010. A multicenter phase II study of erlotinib and sorafenib in chemotherapy-naive patients with advanced non–small cell lung cancer. Clinical Cancer Research, 16(11), pp.3078-3087.

[56] Applegate, C.C. and Lane, M.A., 2015. Role of retinoids in the prevention and treatment of colorectal cancer. World journal of gastrointestinal oncology, 7(10), p.184.

[57] Brown, G.T., Cash, B.G., Blihoghe, D., Johansson, P., Alnabulsi, A. and Murray, G.I., 2014. The expression and prognostic significance of retinoic acid metabolising enzymes in colorectal cancer. PloS one, 9(3).

[58] Bhattacharya, N., Yuan, R., Prestwood, T.R., Penny, H.L., DiMaio, M.A., Reticker-Flynn, N.E., Krois, C.R., Kenkel, J.A., Pham, T.D., Carmi, Y. and Tolentino, L., 2016. Normalizing microbiota-induced retinoic acid deficiency stimulates protective CD8+ T cell-mediated immunity in colorectal cancer. Immunity, 45(3), pp.641-655.

[59] NTP (National Toxicology Program). 2016. Report on Carcinogens, Fourteenth Edition.; Research Triangle Park, NC: U.S. Department of Health and Human Services, Public Health Service

[60] Mohr, U., Haas, H. and Hilfrich, J., 1974. The carcinogenic effects of dimethylnitrosamine and nitrosomethylurea in European hamsters (Cricetus cricetus L.). British journal of cancer, 29(5), pp.359-364.

[61] Mahajan, R., Blair, A., Lynch, C.F., Schroeder, P., Hoppin, J.A., Sandler, D.P. and Alavanja, M.C., 2006. Fonofos exposure and cancer incidence in the agricultural health study. Environmental health perspectives, 114(12), pp.1838-1842.

[62] Ying, R., Bourgeois, D., You, J., Zitnik, M. and Leskovec, J., 2019. Gnn explainer: A tool for post-hoc explanation of graph neural networks. arXiv preprint arXiv:1903.03894.

[63] Yuan, H., Tang, J., Hu, X. and Ji, S., 2020. XGNN: Towards Model-Level Explanations of Graph Neural Networks. arXiv preprint arXiv:2006.02587.

[64] Li, H., Yu, L., Li, M., Chen, X., Tian, Q., Jiang, Y. and Li, N., 2020. MicroRNA-150 serves as a diagnostic biomarker and is involved in the inflammatory pathogenesis of Parkinson's disease. Molecular Genetics Genomic Medicine, 8(4), p.e1189.

[65] Dong, T.N.N. and Khosla, M., 2020. A consistent evaluation of miRNA-disease association prediction models. bioRxiv.

[66] Khurana, P., Gupta, A., Sugadev, R., Sharma, Y.K. and Kumar, B., 2020. HAHmiR. DB: A Server Platform For High Altitude Human miRNA-Gene Coregulatory Networks And Associated Regulatory-Circuits. bioRxiv.

[67] Pham, V.V.H., Liu, L., Bracken, C., Nguyen, T., Goodall, G., Li, J. and Le, T., 2020. pDriver: A novel method for unravelling personalised coding and miRNA cancer drivers. bioRxiv.

[68] Keiser, M.J., Roth, B.L., Armbruster, B.N., Ernsberger, P., Irwin, J.J. and Shoichet, B.K., 2007. Relating protein pharmacology by ligand chemistry. Nature biotechnology, 25(2), pp.197-206.

[69] Cheng, A.C., Coleman, R.G., Smyth, K.T., Cao, Q., Soulard, P., Caffrey, D.R., Salzberg, A.C. and Huang, E.S., 2007. Structure-based maximal affinity model predicts small-molecule druggability. Nature biotechnology, 25(1), pp.71-75.