2024

Machine Learning Algorithms to Study Multi-Modal Data for Computational Biology

Khandakar Tanvir Ahmed
University of Central Florida

MACHINE LEARNING ALGORITHMS TO STUDY MULTI-MODAL DATA FOR COMPUTATIONAL BIOLOGY

by

KHANDAKAR TANVIR AHMED
M.S. University of Central Florida, 2022
B.S. Bangladesh University of Engineering and Technology, 2017

A dissertation submitted in partial fulfilment of the requirements
for the degree of Doctor of Philosophy
in the Department of Computer Science
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Spring Term
2024

Major Professor: Wei Zhang

ABSTRACT

Advancements in high-throughput technologies have led to an exponential increase in the generation of multi-modal data in computational biology. These datasets, comprising diverse biological measurements such as genomics, transcriptomics, proteomics, metabolomics, and imaging data, offer a comprehensive view of biological systems at various levels of complexity. However, integrating and analyzing such heterogeneous data present significant challenges due to differences in data modalities, scales, and noise levels. Another challenge for multi-modal analysis is the complex interaction network that the modalities share. Understanding the intricate interplay between different biological modalities is essential for unraveling the underlying mechanisms of complex biological processes, including disease pathogenesis, drug response, and cellular function. Machine learning algorithms have emerged as indispensable tools for studying multi-modal data in computational biology, enabling researchers to extract meaningful insights, identify biomarkers, and predict biological outcomes.

In this dissertation, we first propose a multi-modal integration framework that takes two interconnected data modalities and their interaction network to iteratively update the modalities into new representations with better disease outcome predictive abilities. The deep learning-based model underscores the importance and performance gains achieved through the incorporation of network information into integration process. Additionally, a multi-modal framework is developed to estimate protein expression from mRNA and microRNA (miRNA) expressions, along with the mRNA-miRNA interaction network. The proposed network propagation model simulates in-vivo miRNA regulation on mRNA translation, offering a cost-effective alternative to experimental protein quantification. Analysis reveals that predicted protein expression exhibits a stronger correlation with ground truth protein expression compared to mRNA expression. Moreover, the effectiveness of integrative models is contingent upon the quality of input data modalities and the completeness

of interaction networks, with missing values and network noise adversely affecting downstream tasks. To address these challenges, two multi-modal imputation models are proposed, facilitating the imputation of missing values in time series data. The first model allows the imputation of missing values in time series gene expression utilizing single nucleotide polymorphism (SNP) data for children at high risk of type 1 diabetes. The imputed gene expression allows us to predict the progression towards type 1 diabetes at birth with six years prediction horizon. Subsequently, a follow-up study introduces a generalized multi-modal imputation framework capable of imputing missing values in time series data using either another time series or cross-sectional data collected from the same set of samples. These models excel at imputation tasks, whether values are missing randomly or an entire time step in the series is absent. Additionally, leveraging the additional modality, they are able to estimate a completely missing time series without prior values. Finally, to mitigate noise in the interaction network, a link prediction framework for drug-target interaction prediction is developed. This study demonstrates exceptional performance in cold start predictions and investigates the efficacy of large language models for such predictions.

Through a comprehensive review and evaluation of state-of-the-art algorithms, this dissertation aims to provide researchers with valuable insights, methodologies, and tools for harnessing the rich information embedded within multi-modal biological datasets.

ACKNOWLEDGMENTS

I would like to express my heartfelt gratitude to my Ph.D. advisor, Dr. Wei Zhang, for his unwavering supervision and support throughout my doctoral journey. Additionally, I extend my sincere appreciation to Dr. Haiyan Hu, Dr. Qian Lou, Dr. Wencai Zhang, and Dr. Yanjie Fu for their insightful contributions as members of my committee during the dissertation proposal and defense process.

Furthermore, I wish to extend my thanks to all the collaborators whose invaluable contributions, insights and discussions have significantly improved my works and publications. I am also thankful to Dr. Farhad Hossain who first introduced me to research during my undergraduate studies and paved the way for me to go forward. My gratitude goes to the past and present members of the Zhang Lab, Dr. Jiao Sun, Naima Ahmed Fahmi, Qibing Jiang, Sudipto Baul, and Istiak Ansari Rakib. Our meetings and discussions were a source of continuous inspiration and ideas for me.

Finally, I am especially grateful to my loving parents and siblings for their constant support, encouragement, and sacrifices without which I would not have reached this milestone in my life.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

Multi-omics data holds immense importance in contemporary biomedical research due to its ability to provide a comprehensive understanding of complex biological systems. By integrating information from multiple omics layers, such as genomics, transcriptomics, proteomics, metabolomics, and epigenomics, researchers gain a more holistic view of biological processes and disease mechanisms [2, 3, 4, 5]. This multi-dimensional approach enables the identification of intricate molecular interactions, biomarkers, and pathways that may be missed when studying individual omics layers in isolation [6, 7, 8]. Multi-omics data also facilitates the discovery of novel disease biomarkers, therapeutic targets, and personalized treatment strategies by uncovering hidden patterns and correlations across different molecular levels [9, 10, 11, 12, 13, 14, 15]. Furthermore, the integration of multi-omics data with clinical and phenotypic information enhances our ability to predict disease risk, prognosis, and treatment response with greater accuracy, specifically for heterogeneous diseases [16, 17, 18]. Overall, multi-omics data holds the potential to revolutionize precision medicine and usher in a new era of personalized healthcare by providing deeper insights into the molecular underpinnings of health and disease.

Several advanced multi-omics data integration frameworks have been developed [19, 20, 21, 22]. However, few approaches link different omics profiles using molecular interaction [23]. Most of them ignore the relations across different biological layers in their analysis. To address this issue, we proposed a generative adversarial network-based framework, omicsGAN that can incorporate the interaction information in the multi-omics integration. We aimed to answer whether the inclusion of the interaction network improves the predictive ability of the integrated datasets. Furthermore, we investigated the effect of network integrity on the quality of the integrated data and consequently the downstream tasks. The model was evaluated on breast cancer, lung cancer, and ovarian cancer datasets and showed significant gain in performance over the baselines. We de-

signed another study to validate the use of network-based multi-omics integration. We estimated protein abundance in a cell from mRNA and miRNA expression and our estimated protein expression showed higher correlation with ground truth protein expression. However, these models were adversely affected by the availability and integrity of network information. Data availability is defined by the rate of missing values in the input data. Longitudinal studies, specially biological studies often suffer from high rate of missing values. Our multi-omics type 1 diabetes study had higher than 98% missing values in the time series gene expression. Noisy interaction network is also unavoidable in many scenarios as yet to be discovered connections are prevalent in real life. Therefore, we proposed new studies to address the limitations of data quality and network integrity to ultimately increase the ceiling of multi-omics integrative models.

In the following section, we provide a comprehensive literature review of our three proposed tasks, multi-modal integration, missing value imputation, and link prediction. First, we introduce state-of-the-art multi-modal integration models followed by the existing models for missing value imputation and drug-target interaction prediction. The current challenges and limitations of these tasks are also included in the description. In chapter 3, we present our proposed network-based multi-omics integration framework that overcomes many of the limitations of existing frameworks. In chapter 4, we use a multi-omics approach to estimate protein abundance to show the utility of network-based multi-omics integration. In chapter 5 and 6, we proposed missing value imputation frameworks and evaluated its performance on participants of a type 1 diabetes study. This missing value imputation technique can offer crucial advantage for to multi-omics integration frameworks. Chapter 7 presents a drug-target interaction prediction framework to represent our multi-modal link prediction. lastly, in chapter 8, we draw our concluding remarks and discuss the potential for future works on this studies.

CHAPTER 2: LITERATURE REVIEW

2.1 Multi-Modal Integration Models

In this dissertation, we primarily focus on the omics data modalities. Integration of bulk multi-omics data through network-driven approaches can be classified according to the nature of the network employed. We delineate three main network types based on their structural characteristics: intra-omics, inter-omics, and mixed interaction networks. In intra-omics frameworks, a single homogeneous network is built from a specific type of omics data. INF [11] and MoGCN [24] the interaction networks from pairwise similarity of samples within the omics data that are fused together on early stage of the framework. MOGLAM [9], MOGONET [25], SUPREME [26] utilize similarity-based interaction networks for each omics data, leveraging graph neural networks to process both the omics data and their respective interaction networks separately. These frameworks then integrate the learned embeddings for each omics using various late-stage integration schemes. Additionally, some frameworks incorporate known interaction networks, such as protein-protein networks [10]. Early-stage integration involves the fusion of network information prior to the generation of embeddings within the framework, often employing similarity network fusion [8]. Late-stage integration typically utilizes graph convolutional networks and integrates the data in the embedding space. In inter-omics frameworks, a heterogeneous interaction network is constructed between the features of multiple omics datasets without intra-omics connections. In contrast, mixed networks contain both intra- and inter-omics connections within the multi-omics heterogeneous interaction network.

MOGLAM [9] uses GCN and attention mechanism to integrate multi-omics data for a specific task. Representations for each omics is obtained using dynamic graph neural network with feature selection (FSDGCN) that can adjust the graph structure based on the classification performance.

Weighted cosine similarity is used to build up the graph. An attention mechanism is employed to obtain the importance of embedding information in different omics and merge information from all omics into the downstream task. Shafi et al. [10] proposed a statistical framework to integrate multi-omics datasets. They identify differentially expressed genes (DEGs) and differentially methylated genes (DMGs) from two p-values, one from classical hypethesis testing and another from effect size estimation. DEGs, DMGs, and known protein-protein interaction (PPI) network are solved for maximum clique problem to identify functional subnetwork. INF [11] proposed a ranked similarity network fusion (rSNF) based approach to select features to train a classifier using multi-omics data. First, a classifier is trained on the juxtaposed multi-omics data, ranking features by ANOVA F-value (juXT). A classifier is again trained on the juxtaposed dataset, with rSNF to rank the features. Finally, the classifier is trained on the juxtaposed dataset restricted to the intersection of top discriminant features from the juXT and rSNF pipelines. MDICC [12] uses sample affinity matrices to represent the network in the multi-omics data. An affinity matrix is constructed for each omics data using euclidean distance and K nearest neighbors algorithm (KNN). The omics specific networks are fused into a single network to integrate the available information and reduce the impact of noise on the original networks. The integrated affinity matrix is then clustered using K-means++ for survival analysis and identification of biomarkers. MoGCN [24] proposes a GCN based framework where the features are generated by an autoencoder from the multi-omics data. The multi-omics autoencoder have multiple encoders and decoders that share a common latent layer. Therefore, the latent representation in the autoencoder contain information from all omics. Euclidean distance based similarity matrix is constructed for each omics and fused together using SNF. The fused network acts as the adjacency matrix in the GCN along with the combined features to classify patients. MOGONET [25] also proposes a GCN based multi-omics integration framework. They construct a weighted sample similarity network for each type of omics data using cosine similarity. An individual GCN works on each omics data to predict the labels for omics specific learning. View Correlation Discovery Network (VCDN) at the label space is then utilized for

multi-omics integration and predict the final labels. Wang et al. [27] uses a feature -level fusion and a network-level fusion to integrate multi-omics data. First they construct patient similarity network (PSN) based on Pearson's correlation coefficients between patients. Feature vector for each node in the PSNs is generated using spectral clustering and Stochastic Block Model (SBM) clustering. Feature-level fusion is achieved by concatenating the feature vectors from individual datasets. SNF is employed to achieve Network-level fusion followed by generation of feature vectors from the fused PSN. Feature vectors from network-level fusion and feature-level fusion are used in a deep neural network to predict clinical outcome. SUPREME [26] builds up similarity matrices based on Pearson's correlation for each omics data. GCNs are used to generate omics specific embedding. They choose different combination of omics and concatenate them to be used in the final prediction of cancer sub types.

2.2 Missing Value Imputation

In this dissertation, we focus on the imputation of missing values in time series data. Handling techniques of missing values in a time series data can be broadly divided into two classes. The first class is case deletion where incomplete observations are removed from the analysis [28]. This is a useful approach if the missing rate is low. As the missing rate increases, case deletion presents a significant drawback by ignoring important information in deleted data. The second approach is imputing the missing value with a reasonable estimation. It can be simple imputation methods such as mean imputation, median imputation, and last observation imputation. However, these techniques fail to utilize temporal information as well as capture the relation among features of the same observation in the time series data. There are also more advanced machine learning-based algorithms for missing value imputation. e.g. KNN based imputation [29], Matrix Factorization-based imputation [30], and maximum likelihood Expectation-Maximization (EM) based imputa-

tion [31]. Although they can capture relations among features, they still cannot exploit temporal information. Recently, deep learning-based imputations, powered by recurrent neural networks, and generative adversarial networks have shown remarkable success in estimating missing values due to their ability to interpret temporal dependency in data and map complex relations among features [32, 33].

Existing studies for time series imputation are uni-modal and self-imputation where the missing values are imputed only using the available values in the same dataset [34, 35]. However, the real world is filled with multi-modal time series data that is being increasingly used in studies [36], thanks to the advancement in data collection and processing technologies. Generally, data from different modalities contain complementary information [37, 38] and the introduction of this complimentary information can further improve the missing value estimation over existing self-imputation models. Multi-modal imputation for cross-sectional data has already shown success [39] which can also be extended to the time series domain. Nonetheless, multi-modal time series imputation comes with some unique challenges. The first challenge is, one of the data can be cross-sectional which means we need a model that can effectively map cross-sectional data to another time series. The second challenge is that some samples can have no available time series data. This may happen if the cross-sectional data is collected for a larger population compared to the time series data due to expensive and logistically difficult data collection [40]. Multi-modal imputation can help us estimate the data for these completely missing samples which is by default not possible in uni-modal imputation techniques.

2.3 Drug-Target Interaction Prediction

This dissertation focuses on drug-target interaction (DTI) prediction as a representation of the link prediction problem. Over the years, various computational approaches, including machine

learning algorithms, network-based methods, and molecular docking simulations, have been utilized for DTI prediction with demonstrated efficacy. Recent advancements in DTI prediction have been accelerated significantly, owing to the extensive accumulation and accessibility of biomedical datasets. This surge has been further fueled by the remarkable progress of deep learning techniques, which have proven successful across diverse scientific research domains and have become the predominant method for DTI prediction. These frameworks can be broadly categorized into knowledge graph-based methods [1, 41, 42, 43, 44], 3D structure-based approaches [45, 46, 47, 48, 49], 2D pairwise distance map-based techniques [50, 51], and 1D sequence-based methods [52, 53, 54, 55, 56]. Knowledge graph-based methods have shown success in various DTI prediction scenarios, including warm start and cold start predictions for drugs and proteins. Cold start predictions, particularly involving unknown drugs or proteins, present significant challenges due to limited information during model training. Despite these challenges, knowledge graph-based models utilize semantic relationships with other entities and diverse data sources to achieve competitive performance. Structure and sequence-based methods, on the other hand, tend to perform less effectively for cold start predictions, especially when the cold start protein or drug lacks structural or sequential homologs with known interactions in the training data. Obtaining high-quality structural data for all proteins of interest can be time-consuming and computationally intensive. In contrast, 1D sequences, such as amino acid sequences for proteins and SMILES for drugs, offer readily available input data that require less computation and simpler quality assurance processes. Addressing the limitations associated with cold start problems using 1D sequences holds promise for accurately predicting interactions across a broader spectrum of drugs and proteins compared to other methods. This dissertation aims to explore and address these challenges in DTI prediction, contributing to advancements in drug discovery and personalized medicine.

CHAPTER 3: NETWORK-BASED MULTI-OMICS INTEGRATION

The work in this chapter has been published in the following paper:

Khandakar Tanvir Ahmed, Jiao Sun, Sze Cheng, Jeongsik Yong, and Wei Zhang (2022). "Multi-omics data integration by generative adversarial network." Bioinformatics, 38(1), 179-186. [38]

3.1 Introduction

Complex diseases such as cancer are highly heterogeneous with different subtypes leading to varying clinical outcomes including prognosis, response to treatment, and chances of recurrence and metastasis [16, 17, 18]. Disease phenotype prediction has been the subject of interest to clinicians and patients for many decades. With the advent of sophisticated technologies enabling the simultaneous collection of diverse biological information, researchers are now able to acquire data from various modalities such as genomics, proteomics, metabolomics, and imaging, among others [57]. It has revolutionized medical and biological research by offering a more comprehensive view of the underlying biological process of disease and identify accurate molecular signatures for characterizing or predicting disease phenotypes [58, 59]. Therefore, studying multi-modal data has become increasingly essential in the field of computational biology, primarily due to its ability to provide a comprehensive understanding of complex biological systems [60]. Integrating these diverse datasets has emerged as a crucial strategy for unraveling the intricate relationships and interactions within biological systems, offering unprecedented insights into disease mechanisms, biological processes, and drug responses. Analysis of multi-omics data along with clinical information of patients can help bridging the gap between genotype and phenotype by exploring the

flow of information within different omics layers [6].

For instance, microRNA (miRNA) regulates mRNA expression by complementarily binding to recognition sequences in the 3' untranslated region of their target mRNAs leading to mRNA degradation and/or mRNA translation inhibition [61]. The abundance of a particular miRNA does not illustrate the full picture without knowing which mRNAs get inhibited by that miRNA; because miRNA does not directly influence the phenotype; rather, regulates the mRNA translation into protein that subsequently determines the phenotype. Moreover, mRNA can be regulated by other modulators like RNA binding protein (RBP) [62]. RBPs bind RNA through globular RNA-binding domains (RBDs) and alter the expression of the bound RNAs [63]. RNA-RBP interaction obtained from crosslinking and immunoprecipitation-based CLIP-Seq can also be applied to characterize the relation between omics data. Hence, integrating the interaction network into multi-omics data analysis will capture the regulatory effect and establish a better correlation with the phenotype.

Several advanced multi-omics data integration frameworks have been proposed in the last five years [19, 20, 21, 22]. However, few approaches link different omics profiles using molecular interaction [23]. Most of them ignore the relations across different biological layers in their analysis. The power of high throughput technologies cannot be fully utilized unless the multi-omics data with its intermodal relations are considered in studies.

In recent years, generative adversarial networks (GAN) [64] has gained popularity in solving problems within the scope of computational biology. GANs take random noise or predefined data as input and generate plausible synthetic data similar to a real dataset by imitating the distribution of the real data. There are several studies that use GAN based algorithms to generate data from single or multiple omics datasets. [65] used GAN for better biomarkers identification by generating a reconstructed functional interaction network from multi-omics datasets. [66] integrated diverse single-cell RNAseq (scRNA-seq) datasets from different labs and experimental protocols

to simulate realistic scRNA-seq data that covers the full cell type diversity. [67] on the other hand used GAN to generate gene expression from bulk RNA-seq datasets. GANs can learn non-linear relationships between features of omics data during training that can be used later for additional insight [66]. It can handle missing data and also promising for missing value imputation because of its capability of learning and imitating any distribution of data [68]. Based on its property of imitating distribution, we can design a GAN with one omics data from one distribution as input to the generator and another omics data with different distribution as real dataset in the discriminator to generate a synthetic data retaining information from both omics datasets.

3.1.1 Contribution

In this chapter, we propose a biologically-motivated deep learning-based model, omicsGAN, to predict disease phenotype by integrating two omics data and the interaction between them (e.g., mRNA expression, miRNA expression, and miRNA-mRNA interaction network). The proposed model introduces a generative adversarial method to generate a new enriched feature set for each omics data combining information from the other omics dataset and the interaction network resulting in a better prediction. Experimental results verify that our proposed framework generates datasets with stronger molecular signatures to better understand the biological mechanism that leads to the disease state and improve disease outcome prediction compared to the biological features derived from single or concatenated omics data.

3.2 Methods

In this section, we first introduce the mathematical notations employed in this study, followed by the proposed framework, omicsGAN, for generating synthetic omics data for disease outcome

prediction using multi-omics data. The framework can take any two omics data with biological relations between each other as input. In this section, we used miRNA, mRNA, and miRNA-mRNA interaction network for illustrative purposes. We then discuss the evaluation metrics and introduce two evaluation methods; a classification model and a penalized Cox regression model that use the synthetic data for disease phenotype prediction and patient survival prediction, respectively.

3.2.1 Overview of the Framework

For the multi-omics data analysis, using extra omics data as an independent feature set provides additional information for downstream analysis. However, different omics profiles are often linked with each other through a complex biological interaction network. Our proposed framework, omicsGAN, can capture the information from this inter-omics network and integrate it with the omics datasets through a generative adversarial network to update them iteratively. After successful training of the network, it will generate new feature sets corresponding to each omics data that contain information from both modality and their interaction network. In this section, the framework is introduced on mRNA and miRNA expression datasets; however, this framework can work with any two omics data that are related to each other, given that their interactions are biologically meaningful. mRNA and miRNA expression are correlated to disease phenotype, although, the bipartite interaction network between them can be leveraged to increase the correlation by incorporating miRNA regulation on mRNA translation. mRNAs directly influence phenotype by translating into proteins that control all physiological activities in a cell; however, miRNA binds to mRNA and regulates its translation into protein, thus indirectly controls the phenotype. From a biological point of view, knowing the expression of a miRNA does not provide enough information without knowing the mRNAs that it targets. For an accurate and realistic downstream analysis, realizing the interaction between omics data into calculation is crucial as well as challenging for the researchers.

Table 3.1: Notations for omicsGAN.

Name	Definition
$\boldsymbol{X} \in \mathbb{R}^{m \times n}$	mRNA expression obtained from RNA-seq
$\boldsymbol{Y} \in \mathbb{R}^{p \times n}$	miRNA expression obtained from miRNA-seq
$\boldsymbol{h}_x^{(k)} \in \mathbb{R}^{m \times n}$	intermediate value of mRNA expression in the k^{th} update
$\boldsymbol{h}_y^{(k)} \in \mathbb{R}^{p \times n}$	intermediate value of miRNA expression in the k^{th} update
$\boldsymbol{H}_x^{(k)} \in \mathbb{R}^{m \times n}$	mRNA expression (synthetic) in the k^{th} update
$\boldsymbol{H}_y^{(k)} \in \mathbb{R}^{p \times n}$	miRNA expression (synthetic) in the k^{th} update
$\boldsymbol{Z}_x \in \mathbb{R}^{m \times n}$	final mRNA expression (synthetic), $\boldsymbol{Z}_x = \boldsymbol{H}_x^{(k^*)}$
$\boldsymbol{Z}_y \in \mathbb{R}^{p \times n}$	final miRNA expression (synthetic), $\boldsymbol{Z}_y = \boldsymbol{H}_y^{(k^*)}$
$\boldsymbol{N} \in \{-1, 1\}^{p \times m}$	adjacency matrix of miRNA-mRNA interaction network
$\boldsymbol{D}_X \in \mathbb{R}^{m \times m}$	diagonal matrix: $\boldsymbol{D}_X(i,i) = \sum_j \lvert\boldsymbol{N}(j,i)\rvert$
$\boldsymbol{D}_Y \in \mathbb{R}^{p \times p}$	diagonal matrix: $\boldsymbol{D}_Y(i,i) = \sum_j \lvert\boldsymbol{N}(i,j)\rvert$
$\tilde{\boldsymbol{S}} \in \mathbb{R}^{p \times m}$	normalized adjacency matrix $\tilde{\boldsymbol{S}} = \boldsymbol{D}_Y^{-\frac{1}{2}} \boldsymbol{N} \boldsymbol{D}_X^{-\frac{1}{2}}$

The notations to define the proposed model, omicsGAN, are summarized in Table 3.1. Let $\boldsymbol{N}$ be the adjacency matrix of miRNA-mRNA interaction network and the dimension of the network is $p \times m$, where p is the number of miRNAs and m is the number of mRNAs. The dimensions of the mRNA ($\boldsymbol{X}$) and miRNA ($\boldsymbol{Y}$) expression data are $m \times n$ and $p \times n$ respectively, with n being the number of samples. Updated (synthetic) mRNA ($\boldsymbol{H}_x^{(k)}$) and miRNA ($\boldsymbol{H}_y^{(k)}$) where $k \in \{1, 2, 3, ..., K\}$, will correspond to the dimension of the input mRNA and miRNA expression datasets respectively and K is the total number of updates in omicsGAN.

In this study, we predict disease outcome using two omics data and the interaction network between them as illustrated in Figure 3.1(a). The framework takes mRNA ($\boldsymbol{X}$), miRNA ($\boldsymbol{Y}$), and normalized interaction network ($\tilde{\boldsymbol{S}}$) as input and iteratively updates them to find two new feature sets that incorporates information from both omics data and their biological interactions, where $\tilde{\boldsymbol{S}} = \boldsymbol{D}_Y^{-\frac{1}{2}} \boldsymbol{N} \boldsymbol{D}_X^{-\frac{1}{2}}$. $\boldsymbol{D}_X$ and $\boldsymbol{D}_Y$ are two diagonal matrices with $\boldsymbol{D}_X(i,i) = \sum_j \lvert\boldsymbol{N}(j,i)\rvert$ and

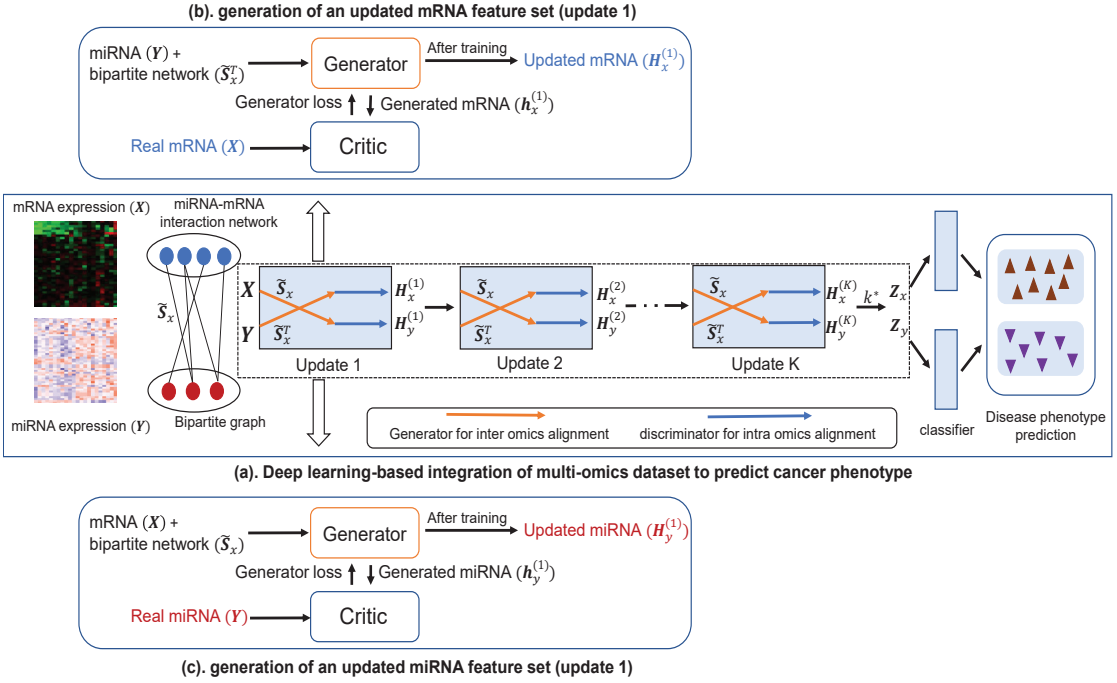

Figure 3.1: (a) An illustration of the proposed generative adversarial framework (omicsGAN). Two omics datasets are updated once in each box through an adversarial game between the generator (marked by orange line) and critic (marked by blue line). Generator and critic are trained for each omics data independently and the updated datasets are applied for disease phenotype prediction. (b) Update of mRNA feature set. Generator uses miRNA expression data and miRNA-mRNA bipartite network to synthesize an mRNA expression data. Both synthetic and input mRNA expression data are passed through a critic that tries to differentiate the real and synthetic data. (c) Update of miRNA feature set. Generator uses mRNA expression data and miRNA-mRNA bipartite network to synthesize an miRNA expression data. Both synthetic and input miRNA expression data are passed through a critic that tries to differentiate the real and synthetic data.

$\boldsymbol{D}_Y(i,i) = \sum_j |\boldsymbol{N}(i,j)|$. A classification model is then applied on the new feature sets to predict the disease phenotype. Figures 3.1(b) and (c) illustrate the frameworks for the first update (k = 1) of the mRNA and miRNA datasets respectively. Each box in Figure 3.1(a) represents k^{th} update which contains two Wasserstein GANs (wGANs) [69] for two omics data. After the wGANs are successfully trained, each generator generates a synthetic data which will be alike the input omics dataset and considered as the updated omics data from that box. For each update, an intermediate

value for miRNA expression is first generated from the generator using mRNA expression and normalized adjacency matrix representing the interaction network. An intermediate value for the mRNA is also found in a similar procedure:

$$\boldsymbol{h}_x^{(k)} = G(\boldsymbol{H}_y^{(k-1)}, \tilde{\boldsymbol{S}}^{\boldsymbol{T}}) \tag{3.1}$$

$$\boldsymbol{h}_y^{(k)} = G(\boldsymbol{H}_x^{(k-1)}, \tilde{\boldsymbol{S}}). \tag{3.2}$$

This mRNA (or miRNA) intermediate value $\boldsymbol{h}_x^{(k)}$ contains information from miRNA (mRNA) in the last update $\boldsymbol{H}_y^{(k-1)}$ and interaction network $\tilde{\boldsymbol{S}}$ but has no relation with the mRNA (miRNA) expression value $\boldsymbol{H}_x^{(k-1)}$ in the last update. The intermediate mRNA (or miRNA) expression value $\boldsymbol{h}_x^{(k)}$ along with the input mRNA (miRNA) expression value $\boldsymbol{H}_x^{(k-1)}$ are then passed through a critic to ensure they are similar to each other:

$$loss_x = D_{loss}(\boldsymbol{h}_x^{(k)}, \boldsymbol{H}_x^{(k-1)}) \tag{3.3}$$

$$loss_y = D_{loss}(\boldsymbol{h}_y^{(k)}, \boldsymbol{H}_y^{(k-1)}) \tag{3.4}$$

D_{loss} is the critic loss between the intermediate value and the input value. After training by minimizing the critic loss, the updated mRNA and miRNA dataset $\boldsymbol{H}_x^{(k)}$ and $\boldsymbol{H}_y^{(k)}$ are learned respectively. This step force the distribution of $\boldsymbol{H}_x^{(k)}$ (or $\boldsymbol{H}_y^{(k)}$) towards the distribution of $\boldsymbol{H}_x^{(k-1)}$ ($\boldsymbol{H}_y^{(k-1)}$). The boxes (updates) in Figure 3.1(a) are arranged in a cascaded structure where each box is trained separately. Once we have trained and got updates $\boldsymbol{H}_x^{(k)}$ and $\boldsymbol{H}_y^{(k)}$ from box k, it is used as input in the following (k+1)th box. $\boldsymbol{H}_x^{(0)} = \boldsymbol{X}$ and $\boldsymbol{H}_y^{(0)} = \boldsymbol{Y}$ are the input to the first layer (box) and after the K^{th} update, $\boldsymbol{Z}_x = \boldsymbol{H}_x^{(k^*)}$ and $\boldsymbol{Z}_y = \boldsymbol{H}_y^{(k^*)}$ are our final synthetic datasets which are used for the disease phenotype prediction, where k^* is the update that gives best prediction result on a separated validation set of samples.

3.2.2 Generative adversarial network model

Generative adversarial network (GAN) models are a class of unsupervised learning task that automatically discovers and learns patterns and distribution in input data in a way that the models can be used to generate new examples that plausibly could have been drawn from the original dataset. It has been widely used in image generation technologies [70]. With some appropriately placed conditions, it can also be used in computational biology to synthesize omics data. In general, GANs use random noise to generate synthetic dataset by requiring the distribution of the random noise towards the distribution of the original data. It does not have to retain information from the random noise; rather, try to make the noise as close to the original data as possible in terms of distribution. In multi-omics study, we can introduce a stream of information from one omics data in place of random noise and incentivize the GAN to retain information from this stream by using appropriate hyperparameters as well as forcing the distribution towards a second omics data. This will ensure the integration of information from both omics data in the generated samples. We can also fuse the interaction network in the model through the generator following the works of [71].

Our proposed pipeline has two separate wGANs for two omics data to update them into a new representation. Generators in each wGAN are three layers fully connected neural network that generates a dataset based on one omics data and the normalized adjacency matrix following the equations:

$$\boldsymbol{h}_x^{(k)} = (ReLU(ReLU(\tilde{\boldsymbol{S}}^{\boldsymbol{T}}\boldsymbol{H}_y^{(k-1)}\boldsymbol{W}^{(0)}))\boldsymbol{W}^{(1)})\boldsymbol{W}^{(2)} \tag{3.5}$$

$$\boldsymbol{h}_y^{(k)} = (ReLU(ReLU(\tilde{\boldsymbol{S}}\boldsymbol{H}_x^{(k-1)}\boldsymbol{W}^{(0)}))\boldsymbol{W}^{(1)})\boldsymbol{W}^{(2)} \tag{3.6}$$

where $\boldsymbol{W^l}$ is the weight matrix in l^{th} layer and rectified linear unit (ReLU) is the activation function. A fully connected neural network is then trained as a critic to assign values to the obtained

intermediate representation $\boldsymbol{h}_x^{(k)}$ and input dataset $\boldsymbol{H}_x^{(k-1)}$. The critic is trained five times for one training of the generator. Objective function for training the critic is:

$$\mathcal{L}_C = C(\boldsymbol{h}_x^{(k)}) - C(\boldsymbol{H}_x^{(k-1)}) \tag{3.7}$$

where C stands for the critic. Critic assigns larger values to the real samples (i.e., $\boldsymbol{H}_x^{(k-1)}$) and smaller values to the synthetic ones (i.e., $\boldsymbol{h}_x^{(k)}$), thus trained by minimizing equation 3.7. On the other hand, generator tries to produce synthetic data that will fool the critic into thinking it as real. Objective function for training the generator is:

$$\mathcal{L}_G = -C(\boldsymbol{h}_x^{(k)}) + \alpha \|\boldsymbol{h}_x^{(k)} - \boldsymbol{X}\|_2 \tag{3.8}$$

where α is a coefficient to control the weight put on the two terms of the equation. For a successful training, generator has to produce data $\boldsymbol{h}_x^{(k)}$ realistic enough that will be assigned a larger value by the critic; therefore, it is trained by minimizing equation (3.8). An L_2-norm is added to further steer the updated dataset towards the original mRNA expression and preserve the feature characteristics. $\boldsymbol{h}_y^{(k)}$ and $\boldsymbol{H}_y^{(k)}$ for miRNA update is derived using analogous equations.

3.2.3 Evaluation methods

3.2.3.1 Classification model

We designed cancer outcome classification tasks with the assumption that better quality of the synthetic datasets will lead to better signatures for disease phenotype prediction compared to the original omics data. Support vector machine (SVM) with linear kernel is implemented as a classifier for all experiments. The datasets are divided into a ratio of 60%, 20%, 20% as numbers of training, validation, and test samples respectively. This model was implemented via Python

package sklearn.svm (SVC)

3.2.3.2 Survival prediction model

A Cox proportional hazards model with Elastic Net penalty [72] is applied to study the correlation between patient's overall survival and omics profiles. The Elastic Net penalty uses a weighted combination of the L_1-norm and L_2-norm penalties by maximizing the following log-likelihood function,

$$\log L(\boldsymbol{\beta}) - \alpha(r \sum_{i=1}^{m} |\beta_i| + \frac{1-r}{2} \sum_{i=1}^{m} \beta_i^2) \tag{3.9}$$

where $L(\boldsymbol{\beta})$ is the partial likelihood of the Cox model, $\alpha \geq 0$ is a hyper-parameter that controls the amount of shrinkage, $r \in [0, 1]$ is the relative weight of the L_1-norm and L_2-norm penalties, and $\beta_i (i \in [1, m])$ represents the coefficient for the i^{th} genomic feature in the omics data. The omics data is randomly splitted into training (80%) and test (20%) sets. Five-fold cross validation is performed on training data to tune the hyper-parameter α. The high risk group and low risk group are determined by the prognostic index ($\boldsymbol{PI}$) on the independent test set. The $\boldsymbol{PI}$ is the linear component of the Cox model, $\boldsymbol{PI} = \boldsymbol{\beta}^T \boldsymbol{X}_{test}$, where $\boldsymbol{X}_{test}$ is the omics profile of the test set, and its risk coefficient was estimated from the Cox model fitted on the training set. The high risk and low risk groups are generated for Kaplan-Meier survival plot by splitting the ordered $\boldsymbol{PI}$ with equal number of samples in each group in the test set. Python package *scikit survival* [73] is applied to implement Cox proportional hazards model with elastic net, and *lifelines* [74] is used for Kaplan-Meier plotting.

3.3 Experiments

We performed experiments on The Cancer Genome Atlas (TCGA) datasets to evaluate the performance of omicsGAN with two different interaction networks (e.g., miRNA-mRNA interaction network and transcription factor (TF)-gene interaction network). In this section, we first describe the datasets and two interaction networks used in experiments. Next we introduce the experimental setup where we explain how to run our proposed model on TCGA data and generate synthetic omics datasets. Lastly, we performed three experiments to evaluate the performance of omicsGAN and the quality of its generated synthetic data: (1) comparing cancer outcome prediction power of the real and synthetic datasets. The comparison was conducted in two ways: classifying clinical variables of cancer patients and number of significant features identified in each dataset; (2) exploring the impact of an accurate interaction network on the prediction power of synthetic datasets; (3) comparing the cancer patient's overall survival prediction using real and synthetic datasets.

3.3.1 Dataset and networks

The proposed framework, omicsGAN, was tested on The Cancer Genome Atlas (TCGA) breast invasive carcinoma (BRCA), lung adenocarcinoma (LUAD), and ovarian serous cystadenocarcinoma (OV) datasets [75, 76, 77]. The RNA-seq mRNA expression and miRNA expression datasets of each cancer type were downloaded from UCSC Xena Hub [78]. For the mRNA expression, the $log2(x+1)$ transformed RSEM normalized count with 20,531 genes was used and for the miRNA expression, the $log2(x+1)$ transformed RPM value with 2,166 miRNAs was used in this study. The clinical information of the three cancer studies was downloaded from cBioPortal [79]. In breast cancer study, we classify the cancer patients based on estrogen receptor (ER+ vs ER-) and triple negative (TN+ vs TN-) status. Triple negative breast cancer patients test negative for all three receptors that are commonly found in breast cancer: estrogen receptors, progesterone receptors,

and excess HER2 protein. For lung cancer and ovarian cancer studies, we classify the patients based on their survival time.

The miRNA-mRNA interaction network was obtained from TargetScanHuman [80]. TargetScanHuman reports effective miRNA-mRNA interactions with context++ model, thereby providing valuable gene-regulatory networks with the miRNA involved. miRNA can bind to mRNA to cause more rapid degradation of the mRNA molecule, therefore reducing the amount of protein translated from that mRNA. A modified adjacency matrix represented the interaction network, where each interaction was valued as -1 to imitate that miRNA negatively regulates the expression of the targeted mRNA and no interaction was valued as 1. The miRNA-mRNA bipartite network contained 163,568 interactions in total. The TF-gene interaction network was downloaded from RegNetwork [81]. The genes present in both lists of TFs and target genes were removed from the list of target genes. The modified bipartite interaction network contained sets of 1053 and 2859 non overlapping genes representing transcription factors and their target genes respectively with 8170 total interactions between them.

3.3.2 Running omicsGAN on the TCGA datasets

To evaluate the proposed generative model on the TCGA omics datasets, we first updated the mRNA and miRNA (or TF and their target gene) expression profiles 5 times ($K = 5$). The generator and critic are fully connected neural networks with two hidden layers for the generator and one for the critic. The generator hidden layers have 512 and 768 neurons respectively whereas the critic hidden layers have 256 neurons. In both generator and critic, the activation function of the hidden layers is ReLU and the output layer is linear. Moreover, hidden layers in critic have dropout with a probability of 0.3. RMSprop optimizer was applied to train both the generator and the critic. Hyperparameters were selected through grid search and details of the hyperparameters used in this

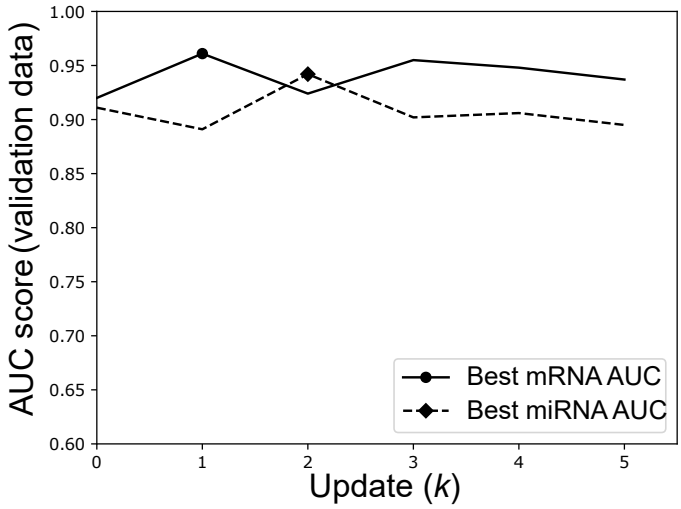

Figure 3.2: Prediction results of triple negative (TN) status on TCGA breast cancer patients using validation samples. AUC of the prediction results using validation samples of synthetic mRNA and miRNA for $k = [1, 2, 3, 4, 5]$. Update k^* with the best validation AUC is selected as the final synthetic data for each omics profile.

study are listed in Table 3.2. In Table 3.2, Omics 1 is the mRNA/gene expression data for both interaction networks, Omics 2 is miRNA expression in miRNA-mRNA interaction network and TF in TF-gene interaction network. The learning rate was chosen from {1e-8, 1e-7, 5e-7, 1e-6, 5e-6, 1e-5, 5e-5, 1e-4, 5e-4, 1e-3, 5e-3, 1e-2} and the candidates for the coefficient α were {1e-5, 1e-4, 1e-3, 1e-2, 0.1, 1, 10}. For batch size, we selected among the options {16, 32, 64, 128, 256}, and no mini batch. The validation set described in the Method section were employed for tuning all hyperparameters. All updated mRNA and miRNA (or gene and TF) datasets ($k = 1, 2, .., 5$) are sequentially fed into the classifier. The support vector machine based classifier described in the Method section was used for classification in all experiments. In the classifier, the dataset was divided into five folds with three folds for training, one fold for validation (parameter tuning and synthetic data update selection), and one fold for testing. We repeated the five-fold splitting 50 times on each dataset. The updated mRNA/gene expression (k^*) with the highest AUC score

for validation samples was selected as the final synthetic mRNA/gene expression output from the model and similarly the updated miRNA/TF expression with the highest AUC score for validation samples was selected as the final synthetic miRNA/TF expression output. Figure 3.2 illustrates the process of selecting the final synthetic mRNA and miRNA datasets from all available updates for TCGA breast cancer patients outcome prediction. $k = 1$ gives the best validation AUC for synthetic mRNA expression whereas $k = 2$ gives the best validation AUC for synthetic miRNA expression. Therefore, mRNA update 1 and miRNA update 2 are used for predicting the test samples and the corresponding results are reported in this study. One synthetic data is generated for breast cancer ER and TN status prediction based on the average validation AUC of the two clinical variables.

Table 3.2: Hyperparameters in omicsGAN used in the study.

Hyperparameter	miRNA-mRNA			TF-gene
	BRCA	LUAD	OV	LUAD
Omics 1 generator learning rate	5e-6	5e-6	5e-6	5e-6
Omics 1 critic learning rate	5e-5	5e-5	5e-5	5e-5
Omics 1 L_2-norm coefficient (α)	0.01	0.01	0.1	0.0001
Omics 2 generator learning rate	5e-6	5e-6	5e-6	5e-6
Omics 2 critic learning rate	5e-5	5e-5	5e-5	5e-5
Omics 2 L_2-norm coefficient (α)	0.001	0.001	0.001	0.001

3.3.3 Integration of mRNA and miRNA expression

We generate the synthetic mRNA and miRNA datasets by integrating the two omcis profiles and their interaction network and assess the quality of the synthetic data through three experiments.

3.3.3.1 omicsGAN improved cancer outcome prediction

To evaluate the quality of the synthetic datasets generated by omicsGAN, we designed cancer outcome prediction and significant predictive signature identification tasks on the TCGA breast cancer, lung cancer, and ovarian cancer datasets under the assumptions: (1) The synthetic datasets learned in omicsGAN consider the expressions in both mRNA and miRNA profiles and the biological interactions between them. So they will provide better predictive signatures compared to mRNA and miRNA expressions. (2) The better predictive signatures will improve the disease phenotype prediction.

Table 3.3: The classification performance on TCGA breast cancer, lung cancer, and ovarian cancer datasets. Average AUC scores of classify cancer patients clinical variables on the synthetic mRNA, miRNA datasets generated from omicsGAN and the original mRNA, miRNA expression datasets. *The difference between the results on the original expression data and the synthetic data is statistically significant (p-value < 0.001).

Input data	Breast cancer		Lung cancer	Ovarian cancer
	ER	TN	Survival time	Survival time
mRNA	0.913	0.91	0.675	0.651
synthetic mRNA (omicsGAN)	0.948*	0.949*	0.733*	0.708*
miRNA	0.878	0.904	0.595	0.627
synthetic miRNA (omicsGAN)	0.945*	0.938*	0.733*	0.721*
mRNA+miNRA	0.905	0.921	0.67	0.658

We ran the classifier with above mentioned five-fold splitting 50 times to select the best synthetic data among the 5 updates based on validation samples and classify the test samples using the selected synthetic data. The average AUC scores of 50 splittings are reported in Table 3.3. There are 185 Estrogen Receptor positive (ER+) and 54 ER negative (ER-) samples, 46 triple negative positive (TN+) and 193 TN negative (TN-) samples in the breast cancer dataset, 95 cancer patients

below the survival time cutoff (< 25 months) and 64 above the cutoff (> 50 months) in the lung cancer dataset as well as 61 cancer patients below the survival time cutoff (< 25 months) and 77 above the cutoff (> 50 months) in the ovarian cancer dataset. Table 3.3 illustrates that the synthetic mRNA and miRNA expression generated by omicsGAN achieved better average classification results than original mRNA and miRNA expression for phenotype predictions across all three cancer types. We also add the baseline where we perform the classification with concatenated miRNA and mRNA expression to see whether addition of more omics data is the reason for the improvement. We can see that concatenated data has similar or better prediction ability compared to the original mRNA and miRNA expression dataset; however, synthetic dataset from omicsGAN always outperforms the concatenated data by a significant margin. This signifies that even though the addition of more omics data improves the outcome prediction performance, omicsGAN relies on the interaction network to generate synthetic data with better predictive signal.

Table 3.4: Number of significant features. Number of significant features between synthetic mRNA, miRNA generated by omicsGAN and the original mRNA, miRNA expression on breast cancer, lung cancer, and ovarian cancer datasets.

Input data	Breast cancer		Lung cancer	Ovarian cancer
	ER	TN	Survival time	Survival time
mRNA	4144	3893	227	133
synthetic mRNA (omicsGAN)	4566	4241	372	142
miRNA	91	91	23	20
synthetic miRNA (omicsGAN)	136	127	58	12

We also evaluated the quality of the original and synthetic datasets by comparing the number of significant features identified in each of them. We performed Student's t-test on the expression datasets with different clinical variables. The number of features with a p-value smaller than 0.001 in each dataset except miRNA expression for lung cancer patients are presented in Table 3.4. p-value cutoff of 0.05 is set for miRNA expression for lung cancer patients as no feature had a

p-value smaller than 0.001 in either the real miRNA expression or the synthetic one. We can see an increased number of significant features in synthetic mRNA compared to the original one for all three cancer types. Synthetic miRNA on the other hand has more significant features for breast cancer and lung cancer, but less for ovarian cancer compared to the original miRNA expression datasets. Therefore, omicsGAN enriches the features of synthetic datasets with better predictive signatures that results into improved cancer outcome prediction.

3.3.3.2 Impact of interaction network on cancer outcome prediction

miRNA expression provides additional predictive signals for cancer outcome prediction on top of the mRNA expression; therefore, integrating them into a new feature set will contain more information compared to mRNA and miRNA expression individually. Table 3.3 and 3.4 already illustrates the ability of omicsGAN to improve the cancer outcome prediction performance. However, we hypothesized that omicsGAN harnesses the information of biological interaction between two omics layers from multi-omics interaction network to generate the synthetic datasets with better predictive signals. Hence, we want to investigate whether the improvement in performance is because of the additional omics data or the model can exploit the interaction network for data integration. We design an experiment to explore the effects of the interaction network on synthetic omics data and their predictive performance where we ran the framework 10 times with same settings and input $\boldsymbol{X}$ (mRNA expression), $\boldsymbol{Y}$ (miRNA expression) as before but a different interaction network on TCGA lung cancer datasets. We replaced the true network with 10 different randomized networks with same density as the true one.

miRNA expression provides additional predictive signals for cancer outcome prediction on top of the mRNA expression; therefore, integrating them into a new feature set will contain more information compared to mRNA and miRNA expression individually. Table 3.3 and 3.4 already illustrates

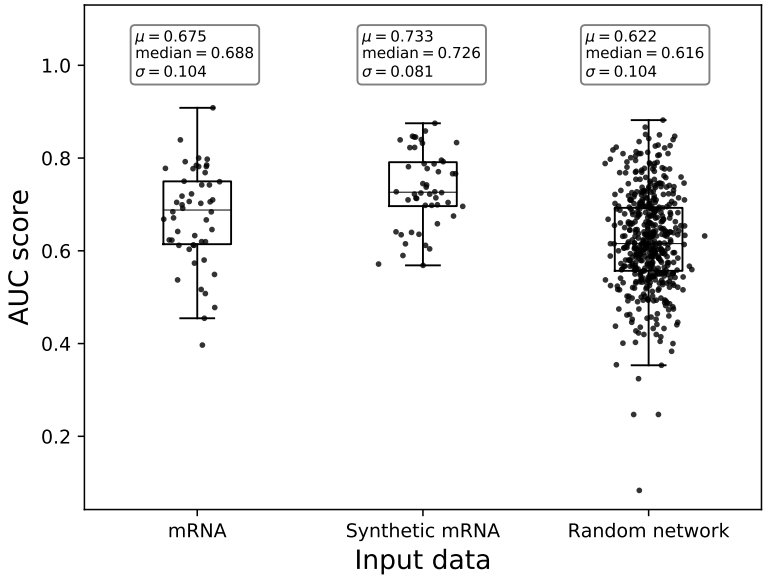

Figure 3.3: Prediction results of the survival time on TCGA lung cancer patients using original and synthetic mRNA expression.Prediction results using original mRNA expression, synthetic mRNA expression generated using true interaction network, and synthetic mRNA expression generated using random interaction network are plotted respectively. Each dot represents the AUC score from one splitting. The statistics (mean, median, and standard deviation) of the prediction performance of the 50 splittings are shown above each boxplot.

the ability of omicsGAN to improve the cancer outcome prediction performance. However, we hypothesized that omicsGAN harnesses the information of biological interaction between two omics layers from multi-omics interaction network to generate the synthetic datasets with better predictive signals. Hence, we want to investigate whether the improvement in performance is because of the additional omics data or the model can exploit the interaction network for data integration. We design an experiment to explore the effects of the interaction network on synthetic omics data and their predictive performance where we ran the framework 10 times with same settings and input $\boldsymbol{X}$ (mRNA expression), $\boldsymbol{Y}$ (miRNA expression) as before but a different interaction network on TCGA lung cancer datasets. We replaced the true network with 10 different randomized networks

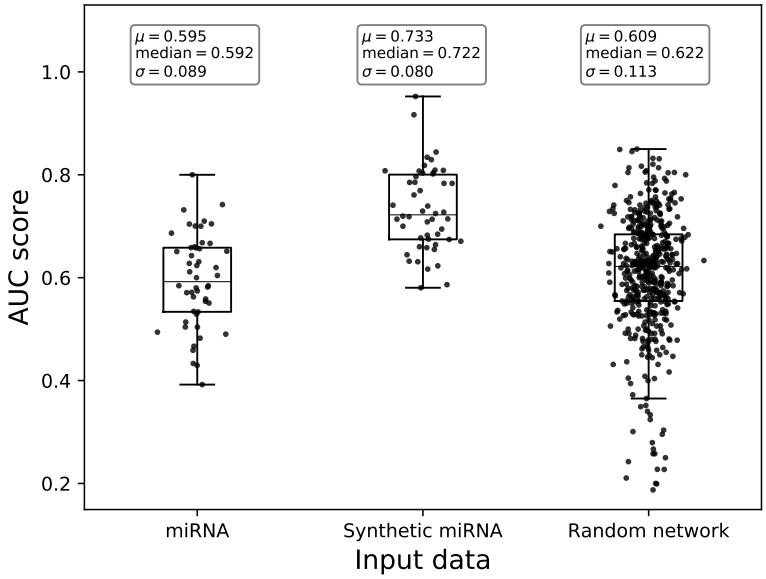

Figure 3.4: Prediction results of the survival time on TCGA lung cancer patients using original and synthetic miRNA expression. Prediction results using original miRNA expression, synthetic miRNA expression generated using true interaction network, and synthetic miRNA expression generated using random interaction network are plotted respectively. Each dot represents the AUC score from one splitting. The statistics (mean, median, and standard deviation) of the prediction performance of the 50 splittings are shown above each boxplot.

with same density as the true one. The prediction results for synthetic mRNA and miRNA expression using true and random networks are shown as boxplots in Figures 3.3 and 3.4 respectively. Prediction results using original mRNA/miRNA expression, synthetic mRNA/miRNA expression generated using the true network, and synthetic mRNA/miRNA expression generated using random network are plotted in each figure. The first two boxplots display the same results for lung cancer outcome prediction as shown in Table 3.3. 50 dots in each of these two boxplots represent the AUC corresponding to 50 random splittings. The third boxplot illustrates the results using 10 random networks, each with 50 splittings. The statistics (mean, median, and standard deviation) of the prediction performance of the splittings are shown above each boxplot. In Figures 3.3 and 3.4, we see a reduction in performance of synthetic mRNA/miRNA expression generated using

a random interaction network compared to the one generated using the true interaction network. This signifies the importance of the interaction network in phenotype prediction and the capability of our framework to capture the information within the network.

3.3.3.3 omicsGAN improved survival prediction

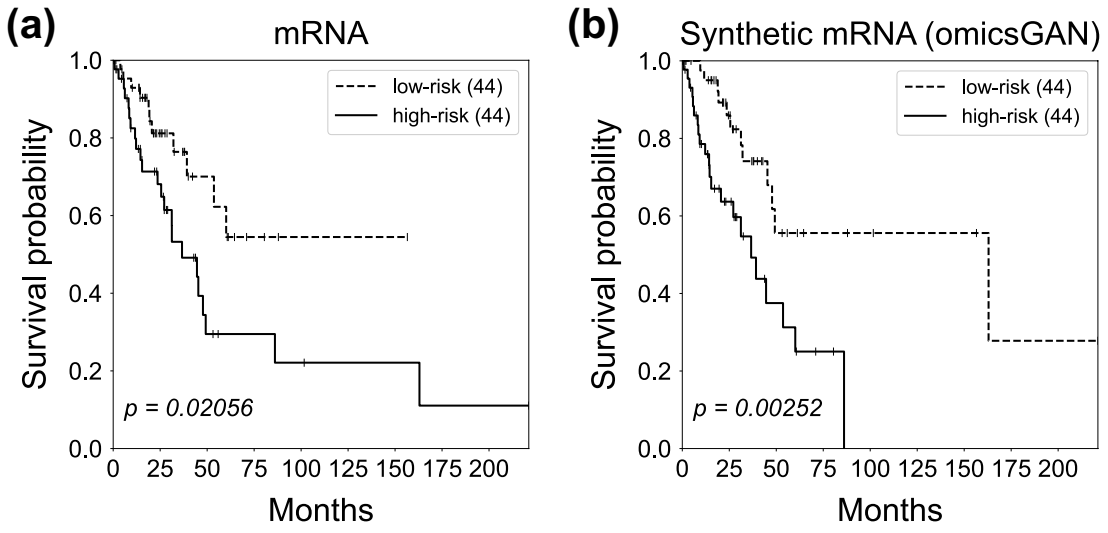

Figure 3.5: Survival prediction on lung cancer patients with mRNA profiles. Kaplan-Meier survival plots for high (solid line) and low (dashed line) risk groups generated by (a) original mRNA, (b) synthetic mRNA expression data on lung cancer patients. The number in the parenthesis indicates the number of samples in low or high risk group. The p-value is calculated by the log-rank test to compare the overall survival of two groups of cancer patients.

To further investigate the quality of the synthetic mRNA and miRNA expression data produced by omicsGAN, the patient's overall survival was predicted on breast cancer, lung cancer, and ovarian cancer datasets. The Cox proportional hazards model with elastic net penalty as described in section 3.2.3 evaluates the correlation between patient's overall survival and genomic features, i.e., the original mRNA, miRNA expressions and the synthetic mRNA, miRNA expressions in this study. The relative weight r in equation 3.9 was set to be 0.5 to combine the subset selection property

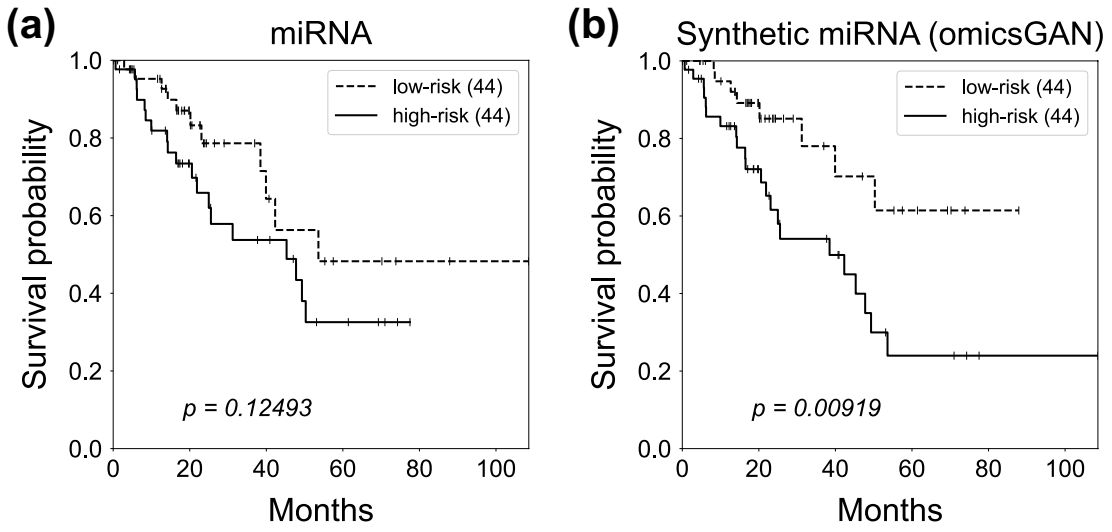

Figure 3.6: Survival prediction on lung cancer patients with miRNA profiles. Kaplan-Meier survival plots for high (solid line) and low (dashed line) risk groups generated by (a) original miRNA, (b) synthetic miRNA expression data on lung cancer patients. The number in the parenthesis indicates the number of samples in low or high risk group. The p-value is calculated by the log-rank test to compare the overall survival of two groups of cancer patients.

of the L_1-norm with the regularization strength of the L_2-norm. 80% of the patient samples were applied to train the model and the performance was tested on 20% test samples. The low and high risk groups on the independent test set were generated based on the prognostic index ($\boldsymbol{PI}$) as mentioned in section 3.2.3. The survival predictions were visualized by Kaplan-Meier plots and compared by the log-rank test p-values. The Kaplan-Meier plots in Figure 3.5 and 3.6 exemplify the improved patient survival predictions on lung cancer using the synthetic mRNA, miRNA expressions generated by omicsGAN compared to the original mRNA, miRNA expressions. The log-rank test p-values clearly demonstrate a strong additional prognostic power of the synthetic omics profiles beyond the original signatures. Similar observations are identified on breast and ovarian cancer patient samples.

3.3.4 Integration of transcription factor and gene expression

The experiments above shows the ability of omicsGAN to generate synthetic data with better predictive power by harnessing the information from miRNA-mRNA interaction network. Here, we design another experiment using transcription factor (TF)-gene interaction network to evaluate whether omicsGAN can show similar improvement in integrating other omics data and their interaction network. We performed the lung cancer phenotype prediction based on the same classification setup as described in section 3.3.3.1 on TFs and their target gene expression datasets. The average AUC scores of 50 splittings are reported in Table 3.5. Both the synthetic TF and target gene expression performed better in classifying the lung cancer patients based on their survival time than the original TF, gene expression, and concatenated TF and gene expression. These findings signify that our proposed framework can work with varying set multi-omics data.

Table 3.5: The classification performance on TCGA lung cancer dataset. Average AUC scores of classification performance between synthetic gene, TF generated from omicsGAN and the original gene, TF expression on lung cancer datasets. *The difference between the results on the original expression data and the synthetic data is statistically significant (p-value < 0.001).

Input data	Lung cancer
gene	0.645
synthetic gene	0.727*
TF	0.656
synthetic TF	0.743*
gene+TF	0.682

3.4 Discussion

Disease phenotype prediction plays a key role in the fight against heterogeneous diseases like cancer. Multi-omics data powered by next generation sequencing technologies has transformed the field of phenotype prediction by providing a broader view of the molecular profiles. Non-redundant predictive signals from multi-omics data make it crucial to develop an efficient and effective framework for multi-omics data integration. However, integrating them as an independent set of features is inadequate as multi-omics data generated for the same set of samples often have an interactive relation among them. Incorporating the interaction network into the analysis will set a flow of information from one omics data to another like the flow within different omics layers in a cell. In most studies, these inter-omics relations are neglected and it is inefficient to predict phenotype using integrated multi-omics data without considering the interactions. Therefore, the integrating of the bipartite interaction network with multi-omics data can result in improved disease phenotype prediction and designing frameworks capable of such integration is gaining importance.

Synthetic data generated from our proposed framework, omicsGAN, shows improvement in prediction performance which illustrates the capability of the model to successfully retain information from multiple omics data and establish a link between them. All synthetic datasets generated in this study with two interaction networks (i.e., miRNA-mRNA and TF-gene) perform better in cancer outcome prediction compared to the original expression datasets; however, the same model using a random interaction network with same density does not perform as good as the synthetic datasets obtained through true network. It signifies that omicsGAN does not fuse information from the two omics data directly; rather functionally incorporate the interaction network into the integration. Synthetic miRNA expression using random interaction network works better than the original miRNA expression (Figure 3.4) but synthetic mRNA using random interaction network does not perform better than original mRNA expression (Figure 3.3). The reason is, without the

true interaction network, omicsGAN can still integrate information from the two omics data to generate synthetic datasets. In that case, the performance of one synthetic data will depend on the additional information received from the other omics data. Synthetic miRNA receives information from mRNA expression, which is significantly better in lung cancer outcome prediction compared to miRNA and thus improves the performance of synthetic miRNA. Synthetic mRNA on the other hand receives information from miRNA that is worse at prediction compared to mRNA and thus results in a decreased performance. An L_2-norm is added in equation 3.8 to ensure the similarity between the updated and original omics data expression; thus allowing the synthetic data to retain feature space properties of the original omics data.

The framework presents an innovative way for multi-omics data integration incorporating their biological interaction. A larger comprehensive study involving more cancer types can draw a better picture of the improvements in phenotype prediction. Although our study was focused on miRNA-mRNA interaction and TF-gene interaction, the same technique can be extrapolated to any two omics data if their interaction network is biologically meaningful. However, to integrate two omics data with different range, distribution, and format (e.g., mutation and gene expression), an extra pre-processing step is necessary to make them compatible. In this study, all missing data is imputed by zero. The prediction performance can be further improved using advanced data imputation frameworks [82, 83, 39] and multi-omics pre-processing methods [84].

3.5 Summary

Thanks to the rapid evolution of high-throughput technologies, abundant genotype data is accruing, which is expected to grow continuously in the era of precision medicine. Because of the complex interactive nature of omics layers, integration of multi-omics data to extract biologically meaningful information of clinical relevance is a challenging task. The promise of multi-omics analysis will

remain unfulfilled unless we can functionally incorporate the inter-omics interaction network into the analysis. In this chapter, we introduced omicsGAN, a generative adversarial network model to effectively integrate the interaction network and the omics datasets into new synthetic data with better predictive signals. We observed that the synthetic data generated from omicsGAN has better discriminative power on cancer outcome classification and cancer patients survival prediction compared to the original omics datasets. Synthetic datasets also contain more significant features that result in better predictive performance. Additionally, we analyzed the effect of interaction network on the quality of synthetic data. Our results show that omicsGAN does not only gather information from two omics datasets; rather functionally incorporate their biological interaction into the integration. Using a random interaction network does not create a flow of information from one omics data to another as efficiently as the true network.

CHAPTER 4: MULTI-OMICS INTEGRATION FOR PROTEIN ABUNDANCE ESTIMATION

The work in this chapter has been published in the following paper:

Khandakar Tanvir Ahmed, Jiao Sun, William Chen, Irene Martinez, Sze Cheng, Wencai Zhang, Jeongsik Yong, and Wei Zhang (2021). In silico model for miRNA-mediated regulatory network in cancer. Briefings in Bioinformatics, 22(6), bbab264. [37]

In last chapter, we proposed a general purpose network-based multi-omics integration framework that can integrate any two interconnected datasets. The integrated data can be used for different downstream tasks as the integration is not task specific. In this chapter, we study task specific multi-omics integration and propose a model to estimate protein expression from mRNA and miRNA expression without the need for wet lab experiments.

4.1 Introduction

Powered by high-throughput transcriptomic technologies, the RNA-seq method can comprehensively profile the transcriptome-wide changes of gene expression in various biological models including cancer cells [85, 86]. Currently, SRA-NCBI [87], the largest public repository for sequencing data, has more than 800,000 human RNA-seq samples and 730,000 mouse RNA-seq samples. These numbers are expected to grow rapidly due to the reduction in the RNA-seq cost per sample and the increased demand for RNA-seq experiments in biomedical research.

Currently, changes in gene expression in the transcriptome are mostly documented by differential

gene/transcript expression analyses. This is based on the assumption that the amount of mRNAs and their corresponding protein are positively correlated in a given biological model. However, in reality, it is becoming evident that the correlation between the level of mRNA and the corresponding protein is weak; recent studies have shown that the correlation between the cellular protein levels and the abundance of their corresponding mRNAs is approximately 0.4, implying that $\sim$40% of the variations in protein abundance can be explained by measuring the changes of mRNA amounts [88]. Consistently, this weak correlation was also found in cancer tissues, and there are findings that question the validity of using the mRNA expression as a way to understand gene expression [89]. The multiple layers of regulatory mechanisms involved in gene expression after transcription is one explanation for this weak correlation. Although the mRNA expression analysis has its own value in understanding gene expression, it does not provide comprehensive information on the proteome. In an attempt to address this discrepancy, some studies [90, 91] have proposed the use of gene specific RNA-to-protein (RTP) conversion factors. This method would allow for the estimation of protein expression from transcriptomic data; however such methods use the same RTP for all samples and therefore fail to realize the difference between different biological contexts leading to false approximations. Consequently, to draw accurate predictions about the proteome based on transcriptomic data, post-transcriptional regulatory mechanisms must be considered.

Post-transcriptional gene regulation includes but is not limited to splicing, polyadenylation, nuclear export, and miRNA-regulated translation. Numerous bioinformatics pipelines are available to profile post-transcriptional events such as alternative splicing and alternative polyadenylation (APA). Particularly, APA can occur in the 3'-untranslated region (3'-UTR) of mRNAs and can produce an mRNA isoform with a different 3'-UTR length. Recent studies found that more than 70% of the human genes have the capacity to produce 3'-UTR APA isoforms, suggesting the prevalence of APA in the 3'-UTR [92]. Although APA in the 3'-UTR does not affect the coding capacity of a

gene, this region contains binding sites for post-transcriptional regulatory mechanisms (e.g. miRNAs). Therefore, APA in the 3'-UTR potentially affects the mRNA stability or protein production [93, 61]. Several studies showed that proliferating or transformed cells favor the expression of mRNAs with shorter 3'-UTRs through APA and lead to the activation of oncogenes [94, 95]. In addition, highly expressed mRNAs in cancer cells feature a shorter 3'-UTR with fewer miRNA-binding sites and exhibit the decrease of miRNA-mediated translational repression [96, 97].

miRNA expression profiles differ between normal tissues and tumors in cancer patients [98, 99]. Recent studies have shown that miRNA can serve as a molecular marker for the early detection of cancer [100, 101, 102]. Therefore, it is important to investigate how miRNAs post-transcriptionally regulate gene expression in cancer. However, as the cancer transcriptome data and the miRNA expression data are available through high-throughput sequencing, the gene regulatory mechanism of miRNA can only be predicted using miRNA-mRNA interaction modeling. Three miRNA-mRNA interaction databases were built up recently [103, 104, 105] and they provide the positional information for each miRNA-mRNA interaction in the 3'-UTR. However, considering the dynamic regulation of 3'-UTR length by APA in cancer or perturbed cells, a simple one-dimensional mapping of miRNA-mRNA interaction based on the annotated gene structure may not provide a comprehensive picture of post-transcriptional regulation of mRNAs in cancer studies. In addition, the current competing endogenous RNA (ceRNA) model largely ignores the dynamics of 3'-UTR landscape for miRNA-binding sites caused by 3'-UTR APA [106].

4.1.1 Contribution

In this chapter, we present a biologically motivated graph-based learning model, PTNet, to predict the protein expression by integrating the mRNA expression, the miRNA expression, the miRNA-mRNA interaction network, and the dynamics of 3'-UTR in the transcriptome. The proposed

model harnesses the mRNA and miRNA expression in cancer studies and can be applied to existing big data to predict the protein expression; it eliminates the need for a large-scale proteomics experiment. The experimental results confirm that our proposed framework provides a higher resolution of molecular signatures to better understand biological mechanisms that lead to the disease state. Our model also improves a cancer outcome prediction compared to the prediction made by considering the mRNA or miRNA expression only. An advanced deep learning method that integrates the mRNA and miRNA expression data through a controlled fusion layer is also proposed as a baseline method to compare the cancer outcome prediction performance to the proposed graph-based learning model.

4.2 Method

In this section, we first introduce a graph-based learning model, PTNet, which is motivated by miRNA-mediated regulation of gene expression to estimate the level of the corresponding protein. We also introduce the strategies to evaluate the quality of the estimated protein expression. Next, a deep learning-based fusion network model is introduced as a baseline method that integrates multi-omics data (i.e., mRNA and miRNA in this study) to predict patient outcome. This model considers the relation between the biological features within the same omics and across different omics profiles by the fusion network.

4.2.1 PTNet: Graph-based learning model

4.2.1.1 miRNA-mRNA interaction and miRNA-mediated gene regulation

To estimate the protein expression from mRNA expression data, we first accessed the well-established miRNA-mRNA interaction database TargetScan [80] and collected the position information for all

possible miRNA-binding sites in the 3'-UTR of target mRNAs. To establish the miRNA-mRNA interaction network, a miRNA was connected to the expressed mRNAs that contain the binding site in their 3'-UTRs. In the miRNA-mRNA interactive bipartite network, an interaction was valued as -1 to imitate the miRNA induced silencing on target mRNA while no interaction was valued as 1. However, this scoring is neglecting the scenario in which mRNA loses miRNA-binding sites due to 3'-UTR APA events. If the miRNA-binding site is located within the lost 3'-UTR, the shorter mRNA will bypass miRNA-mediated inhibitory regulation while the longer isoform will be suppressed in translation.

4.2.1.2 Graph-based learning algorithm

The notations to define the graph-based learning algorithm are summarized in Table 4.1. Let m be the number of mRNAs, and n be the number of miRNAs. The dimensions of the miRNA-mRNA interaction network $\boldsymbol{N}$, mRNA expression data $\boldsymbol{X}$, and miRNA expression data $\boldsymbol{Y}$ are $n \times m$, $m \times k$, and $n \times k$ respectively, with k being the number of samples. Predicted protein expression $\boldsymbol{F}$ corresponds to the dimension of the mRNA expression dataset.

Given the values of mRNA expression $\boldsymbol{X}$, miRNA expression $\boldsymbol{Y}$, and interaction network $\boldsymbol{N}$, we applied a bipartite graph-based learning model PTNet to predict the abundance of protein expression $\boldsymbol{F}$. Let $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{U}, \boldsymbol{E}, \boldsymbol{N})$ denote an undirected bipartite graph, where $\boldsymbol{V}$ and $\boldsymbol{U}$ are two disjoint vertex sets that represent miRNAs and mRNAs. $\boldsymbol{E}$ is a set of edges that stands for the miRNA-mRNA interactions, and $\boldsymbol{N} \in \{-1, 1\}$ is the adjacency matrix of the network. Since the miRNAs negatively regulate the translation of mRNAs, the elements in the interaction network $\boldsymbol{N}$ are either 1 (no connection) or -1 (connected).

For i^{th} sample, the miRNA vertex set $\boldsymbol{V}$ is initialized by the miRNA expression denoted by $\boldsymbol{y}_i$, which is learned from miRNA-seq data. Similarly, the mRNA vertex set $\boldsymbol{U}$ is initialized by the

Table 4.1: Notations for PTNet model

Name	Definition
$\boldsymbol{X} \in \mathbb{R}^{m\times k}$	mRNA expression, $\boldsymbol{X}$ = $[\boldsymbol{x}_1, \boldsymbol{x}_2, ..\boldsymbol{x}_i, .., \boldsymbol{x}_k]$
$\boldsymbol{Y} \in \mathbb{R}^{n\times k}$	miRNA expression, $\boldsymbol{Y}$ = $[\boldsymbol{y}_1, \boldsymbol{y}_2, ..\boldsymbol{y}_i, .., \boldsymbol{y}_k]$
$\boldsymbol{F} \in \mathbb{R}^{m\times k}$	estimated protein expression, $\boldsymbol{F}$ = $[\boldsymbol{f}_1, \boldsymbol{f}_2, ..\boldsymbol{f}_i, .., \boldsymbol{f}_k]$
$\boldsymbol{N} \in \{-1, 1\}^{n\times m}$	adjacency matrix of miRNA-mRNA interaction network
$\boldsymbol{D}_X \in \mathbb{R}^{m\times m}$	diagonal matrix: $\boldsymbol{D}_X(i,i) = \sum_j \lvert\boldsymbol{N}(j,i)\rvert$
$\boldsymbol{D}_Y \in \mathbb{R}^{n\times n}$	diagonal matrix: $\boldsymbol{D}_Y(i,i) = \sum_j \lvert\boldsymbol{N}(i,j)\rvert$
$\boldsymbol{S} \in \mathbb{R}^{n\times m}$	normalized adjacency matrix, $\boldsymbol{S}$ = $\boldsymbol{D}_Y^{-\frac{1}{2}}\boldsymbol{N}\boldsymbol{D}_X^{-\frac{1}{2}}$
$\lambda \in \mathbb{R}_+$	hyper-parameter

mRNA expression denoted by $\boldsymbol{x}_i$, which is learned from RNA-seq data. Vector $\boldsymbol{f}_i$ denotes the protein expression for sample i which we desire to study and is shown in Figure 4.1. We also introduce a vector $\widetilde{\boldsymbol{y}}_i$, which can be considered as the available miRNA expression after mRNA is translated into its corresponding protein. In this context, the cost function over $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{U}, \boldsymbol{E}, \boldsymbol{N})$ is defined as

$$\begin{aligned}\Omega(\boldsymbol{f}_i, \widetilde{\boldsymbol{y}}_i) = ||\boldsymbol{f}_i||^2 + ||\widetilde{\boldsymbol{y}}_i||^2 - 2\boldsymbol{f}_i^T\boldsymbol{S}\widetilde{\boldsymbol{y}}_i \\ + \lambda||\boldsymbol{f}_i - \boldsymbol{x}_i||^2 + \lambda||\widetilde{\boldsymbol{y}}_i - \boldsymbol{y}_i||^2,\end{aligned} \tag{4.1}$$

where $\boldsymbol{S}$ is a normalized adjacency matrix based on $\boldsymbol{N}$ as shown in Table 4.1 and λ is a regularization parameter for balancing the cost terms on the right side of the equation. The first three terms enforce the consistency between the connected vertex pairs in the miRNA-mRNA bipartite graph. They penalize the miRNA-mRNA interaction with a high estimated protein expression but has the available miRNA that can bind to the mRNA to further suppress its translation. The last two terms are fitting terms which keep the estimated protein expression level and the final miRNA ex-

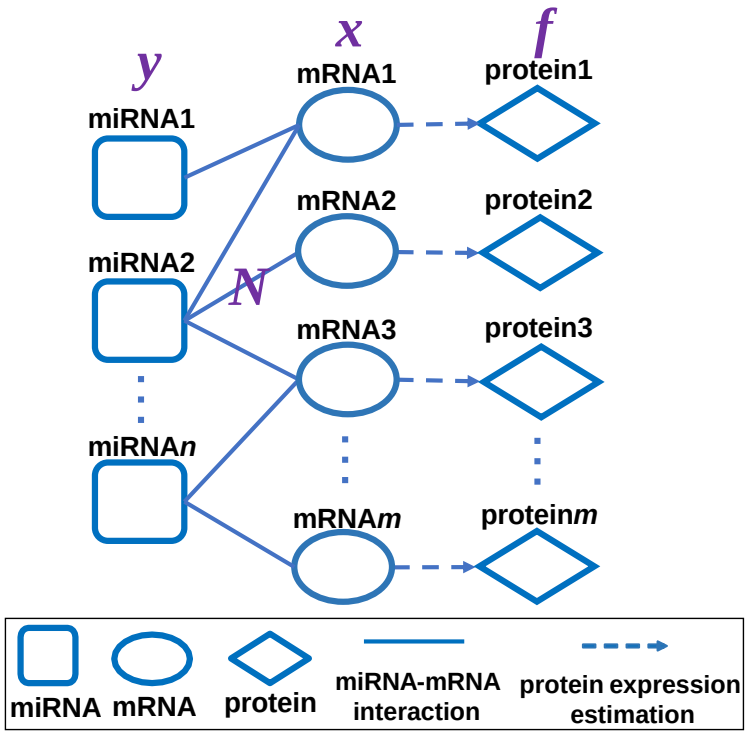

Figure 4.1: An illustration of the proposed graph-based learning model on miRNA-mRNA bipartite graph to estimate the protein expression levels. The miRNA-mRNA interaction networks are built up based on known miRNA binding sites. The miRNA vertex and mRNA vertex are initialized with miRNA expression and mRNA expression, respectively. A graph-based learning model PTNet is applied to imitate the miRNA regulation on the network and to estimate the protein expression levels.

pression level consistent with the initial mRNA expression level and the miRNA expression level, respectively. Similar to the algorithm proposed by [107, 108, 18, 109], the optimization problem in equation 4.1 can be solved with an iterative label propagation algorithm as follow,

$$\boldsymbol{f}_i^t = (1-\alpha)\boldsymbol{x}_i + \alpha \boldsymbol{S}\widetilde{\boldsymbol{y}}_i^{t-1} \tag{4.2}$$

$$\widetilde{\boldsymbol{y}}_i^t = (1-\alpha)\boldsymbol{y}_i + \alpha \boldsymbol{S}^T\boldsymbol{f}_i^{t-1} \tag{4.3}$$

where α = 1/(1+λ), t denotes the propagation iteration, $\widetilde{\boldsymbol{y}}_i^0 = \boldsymbol{y}_i$ and $\boldsymbol{f}_i^0 = \boldsymbol{x}_i$. The label propa-

gation algorithm iteratively performs propagation between the vertices of mRNA and miRNA in both directions as shown in Figure 4.1 and will be converged to a closed-form solution to get the protein expression level. It imitates the post-transcriptional regulation events in cells to capture the protein expression changes due to miRNA regulation.

4.2.2 Evaluation methods

We used two criteria to evaluate the quality of the estimated protein expression proposed by PTNet and compare it to the mRNA expression data and the data resulting from the integration of mRNA and miRNA expression data. First, we measured the consistency between the ground-truth protein expression (proteomics data) and the estimated protein expression or mRNA expression by correlation coefficients. Second, we designed cancer outcome classification tasks with the assumption that a better quality of the protein expression estimation will lead to better molecular signatures for disease phenotype prediction compared to the estimation when only considering mRNA and miRNA expressions.

4.2.2.1 Pearson correlation coefficient

The protein expression was estimated for individual miRNA neighborhood networks by PTNet (equation 4.1). The Pearson correlation coefficient was applied to measure the consistency between the estimated protein expression or mRNA expression and the true protein expression. The formula of Pearson correlation coefficient is defined as

$$r = \frac{\sum_{i=1}^{m}(a_i - \bar{\boldsymbol{a}})(b_i - \bar{\boldsymbol{b}})}{\sqrt{\sum_{i=1}^{m}(a_i - \bar{\boldsymbol{a}})^2}\sqrt{\sum_{i=1}^{m}(b_i - \bar{\boldsymbol{b}})^2}},$$

where $\boldsymbol{a}$ is the estimated protein expression or mRNA expression for one sample and $\boldsymbol{b}$ is the ground-truth. $\bar{\boldsymbol{a}}$ and $\bar{\boldsymbol{b}}$ denote the average expression levels and m is the number of isoforms.

4.2.2.2 Classification model

A feed forward fully connected deep neural network was applied for binary cancer outcome classification on estimated protein expression, mRNA expression, or the integration of mRNA and miRNA expressions. The cost function of the deep learning model is

$$\mathcal{L} = -\boldsymbol{h} log(\boldsymbol{p}) - (1 - \boldsymbol{h}) log(1 - \boldsymbol{p}) \tag{4.4}$$

where $\boldsymbol{h}$ is the truth label of the disease patients and $\boldsymbol{p}$ is the predicted labels. Adam optimizer was used with a learning rate of 0.01. 500 biological features that most correlated with the labels of the training samples were selected as the input for the learning model. This is a two-hidden layer neural network with 250 and 100 neurons in each layer respectively. Both hidden layers use the rectified linear unit (ReLU) as the activation function and the dropout with a probability of 0.2. The output layer uses Sigmoid as the activation function. The area under receiver operating characteristic curve (AUC) score was applied to evaluate the performance of the classifiers and the quality of the input biological features.

4.2.3 Deep learning-based fusion network

The proposed PTNet model considered both mRNA and miRNA expressions in the analysis. To evaluate PTNet and make a fair comparison, we also propose a deep learning-based multi-omics feature extraction framework that considers the relations between different multi-omics features (i.e., mRNA expression and miRNA expression) for a disease outcome prediction as a baseline

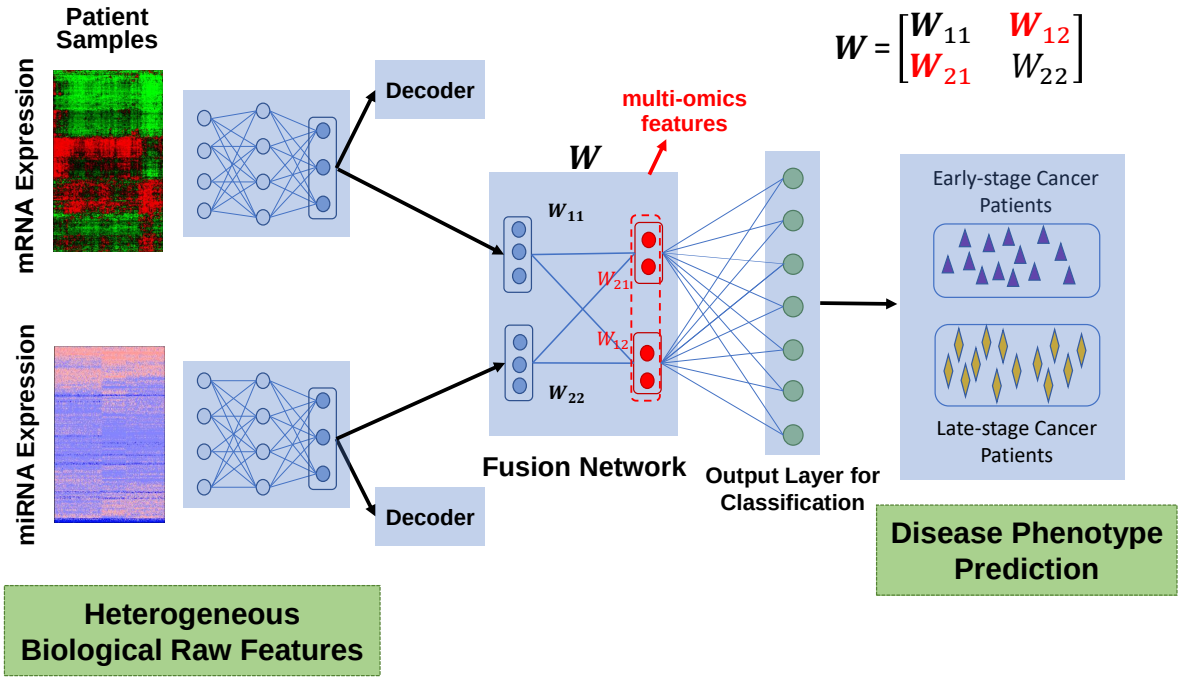

Figure 4.2: Overview of the deep learning-based fusion network model. One autoencoder is constructed for each omics profiling data (left panel). Then, a fused network is learned across the outputs from multi-omics data to identify important multi-omics features (red nodes). Next, the fused multi-omics features are applied for disease phenotype prediction. The structure of network parameters $\boldsymbol{W}$ in the fusion network is shown at the top right corner.

method (Figure 4.2). In this framework, one autoencoder for each input omics data is constructed to project the high dimension low sample size omics profile onto a low-dimensional embedding. The encoder encodes the data, whereas the decoder reconstructs the original data. The minimization of weighted reconstruction loss enforces the features learned from the omics profiles to be salient and robust. The autoencoders are designed with a loss function,

$$\mathcal{L} = \frac{\sum_{i=1}^{k} (\boldsymbol{x}_i - \boldsymbol{x}_i^d)}{k}, \tag{4.5}$$

where $\boldsymbol{x}_i$ and $\boldsymbol{x}_i^d$ are the original mRNA expression and reconstructed mRNA expression from the decoder for sample i respectively. k denotes the number of samples. $\boldsymbol{x}_i^d$ is enforced to be as close to the original features as possible so that maximum retention of information in the learned features is ensured. For miRNA expression ($\boldsymbol{Y}$), another autoencoder with the same loss function is applied.

Then, the learned features, $\boldsymbol{X}^e$ and $\boldsymbol{Y}^e$ from each network are transformed into an input layer of a neural network by considering the relations between the extracted features within the same omics profile and across different omics profiles with a controlled fusion technique. Specifically, the network parameter $\boldsymbol{W}$ in the fusion network in Figure 4.2, is learned upon the relation of the features within the mRNA expression data $\boldsymbol{W}_{11}$, the relation of the features between mRNA and miRNA expression data $\boldsymbol{W}_{12}$ and so on. Different blocks in $\boldsymbol{W}$ are weighted by different regularization coefficients λ and α. We apply ℓ_1-regularization on the off-diagonal blocks in $\boldsymbol{W}$ with the assumption that the connections between the features extracted from different omics profiles are sparse. Thus, the loss function for this framework is

$$\begin{aligned}
&\min_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{X}^e, \boldsymbol{Y}^e, \boldsymbol{h}, \boldsymbol{W}, \lambda, \alpha) = ||C(\boldsymbol{X}^e, \boldsymbol{Y}^e, \boldsymbol{W}) - \boldsymbol{h}||_2^2 \\
&+ \lambda_{11}||\boldsymbol{W}_{11}||_F^2 + \lambda_{22}||\boldsymbol{W}_{22}||_F^2 + \alpha_{12}||\boldsymbol{W}_{12}||_F^2 + \alpha_{21}||\boldsymbol{W}_{21}||_F^2 \\
&+ \lambda_{12}||\boldsymbol{W}_{12}||_1 + \lambda_{21}||\boldsymbol{W}_{21}||_1, \qquad (4.6)
\end{aligned}$$

where $\boldsymbol{h}$ is the truth label of patient outcomes. The first term of the loss function is a binary cross-entropy loss whereas the last two terms enforce the desired sparsity of $\boldsymbol{W}$ described above. $\boldsymbol{W}_{11}$, $\boldsymbol{W}_{12}$, $\boldsymbol{W}_{21}$, and $\boldsymbol{W}_{22}$ are submatrices of $\boldsymbol{W}$ that correspond to mRNA-mRNA, mRNA-miRNA, miRNA-mRNA, and miRNA-miRNA interaction in the fusion network respectively. The multi-omics features are the output of the fusion network and two more layers are added after the fusion network for a disease outcome prediction.

In summary, in this method section, a two-step framework towards the phenotype prediction is proposed: (1) learn the features (estimate protein expression) through the graph-based learning model PTNet and (2) predict the disease phenotype using the learned features as input in the classifier as described in the subsection 4.2.2.2. To predict disease outcomes using the mRNA expression, the same classifier is applied without the first step of the framework for comparison. A multi-omics deep learning-based fusion network is proposed to integrate these two steps allowing the mRNA and miRNA expression datasets as input and directly predicting the disease outcome as output. This model also learns new multi-omics features from mRNA and miRNA expression datasets similar to the graph-based learning model using a fusion network without considering the biological interactions between miRNAs and mRNAs ($\boldsymbol{N}$). Therefore, the new multi-omics features are learned from each modality separately, instead of incorporating the knowledge from post-transcriptional regulation.

4.3 Results

In the experiments, we first generated artificial datasets for two biological conditions to test if the PTNet can capture the changes of protein expression by considering the miRNA-mediated regulatory pathway. Next, we performed three experiments on The Cancer Genome Atlas (TCGA) datasets to evaluate the performance of PTNet. The first experiment was to compare the protein expression estimated by PTNet with the proteome data. The second experiment was to evaluate the prediction power of the estimated protein expression on cancer patient outcomes. The last experiment was to show the effects of 3'-UTR APA on the miRNA-mRNA interaction network and the level of protein expression.

4.3.1 Simulation

In this simulation experiment, we generated two artificial miRNA-mRNA bipartite networks which have different interactions due to 3'-UTR APA events between two biological conditions as shown in Figure 4.3(a). Both bipartite networks consist of three miRNAs and four mRNAs. The expression values of those miRNAs and mRNAs in this simulation were randomized but the two conditions were set to maintain the expression value of corresponding RNAs the same. Due to the 3'-UTR APA events between two biological conditions, miRNA3 loses its binding sites on mRNA2 in Condition 1 and mRNA4 in Condition 2 as illustrated in Figure 4.3(a). Theoretically, the expression of protein2 would then increase while the expression of protein4 would decrease due to the reorganization of miRNA3-binding in Condition 1.

Next, we imitate the miRNA regulation based on the neighboring relations of each miRNA and estimate the protein expression changes depending on the mRNA expression, miRNA expression, and their role in post-transcriptional regulation as formulated in equation 4.1. We run PTNet twice, first with the interactions corresponding to the Condition 1 and second with two interactions altered to simulate the Condition 2 in Figure 4.3(a). The expression values of the mRNA2 and the mRNA4 are plotted in Figure 4.3(b). The initial value (iteration 0) of each plot represents the original mRNA expression whereas the final value of the plot is its corresponding estimation of the protein expression. From these experiments, we observed that the final estimated values of the protein expression are lower than their original mRNA values in Condition 1 since both mRNAs are bound by miRNAs. Then in Condition 2, the estimated expression of protein2 decreases further as a new miRNA (miRNA3) binds to the mRNA2, whereas the protein4 expression increases as the mRNA4 is free of miRNA-binding. In Figure 4.3(b), the predicted changes of the protein expression between the two conditions are as we expected. The proposed model can imitate miRNA-mediated regulation of gene expression and predict the corresponding protein expression.

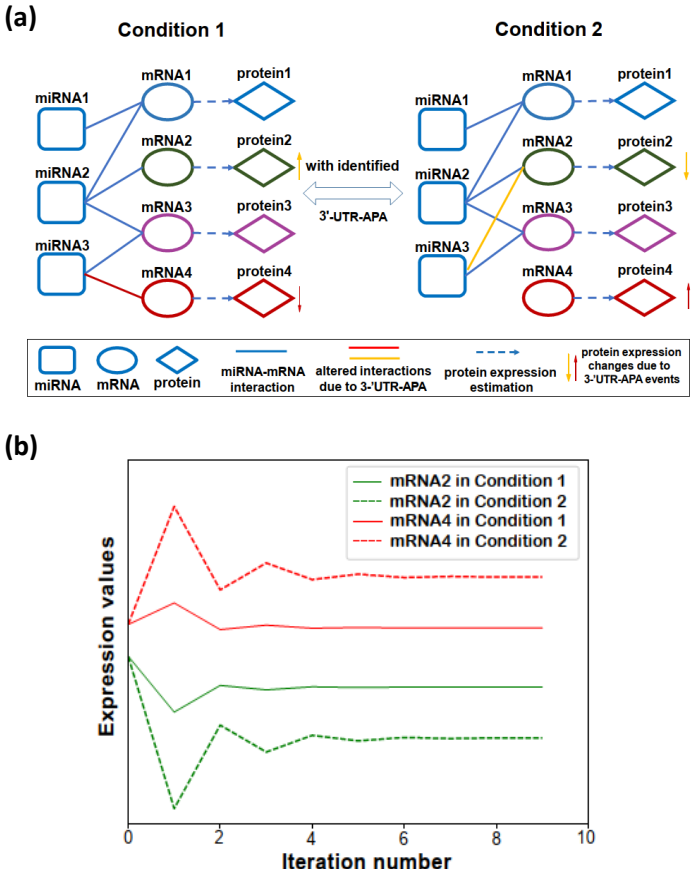

Figure 4.3: (a) Simulated miRNA-mRNA bipartite networks on two biological conditions. The altered interactions due to 3'-UTR APA between two conditions are highlighted as yellow and red lines. The miRNA vertex and mRNA vertex are initialized with miRNA expression and mRNA expression, respectively. (b) The changes of mRNA expression level. The initial value (iteration 0) of each plot represents the original mRNA expression and the final value of the plot is its corresponding estimated protein expression. The iteration number represents the iteration in the label propagation algorithm to solve the optimization algorithm in equation 4.1 as discussed in subsection 4.2.1.2.

4.3.2 Experiments on TCGA datasets

4.3.2.1 Dataset

The proposed graph-based learning model PTNet and the baseline method were tested on TCGA breast cancer (BRCA) and ovarian cancer (OV) datasets [75, 77]. The RNA-seq gene expression and miRNA expression datasets were downloaded from UCSC Xena Hub [78]. For the gene expression, the $log_2(x + 1)$ transformed RSEM normalized count was used in the analyses and 20,531 genes were included in this study. For the miRNA expression, the $log_2(x + 1)$ transformed RPM value was used in the analyses and 2,166 miRNAs were included in this study. The clinical information of the cancer studies was downloaded from cBioPortal [79]. There are 185 Estrogen Receptor positive (ER+) and 54 ER negative (ER-) samples in the breast cancer dataset and 51 cancer patients in the early stage ($\leq$ IIIA) and 359 cancer patients in the late stage ($>$ IIIA) in the ovarian cancer dataset. The protein spectral counts in the proteome data downloaded from National Cancer Institute data portal[1] was used as the ground truth for the protein expression. The miRNA-mRNA interaction network was obtained from TargetScanHuman [80] which predicts effective miRNA target sites within mRNAs. A modified adjacency matrix with 163,568 interactions was applied to represent the network, where each interaction was valued as -1 to imitate the miRNA-mediated negative regulation of targeted mRNAs. No interaction was valued as 1.

4.3.2.2 PTNet improved the estimation of the protein expression

To evaluate the proposed graph-based learning model, we first investigated the effect of an individual miRNA on its neighborhood network and estimated the protein expression of the corresponding mRNAs that bind to the miRNA. The neighborhood network is defined by a targeted miRNA, all

[1] `https://cptac-data-portal.georgetown.edu/cptac/s/S015`

mRNAs directly bound to the targeted miRNA (first-order neighbor of targeted miRNA), and all miRNAs directly connected to the first-order neighbor mRNAs. Interactions between the selected miRNAs and mRNAs from the original interaction network obtained from TargetScanHuman were applied as the interactions in the neighborhood network. We performed a comprehensive literature review of cancer related miRNAs and selected miRNAs that were associated with breast cancer and ovarian cancer pathogenesis (Tables 4.2 and 4.3). We then ran the proposed graph-based learning model to estimate the protein expression for the neighborhood networks. The predicted protein expression was compared to the ground truth spectral count in terms of Pearson correlation coefficients. Detailed results for TCGA breast cancer and ovarian cancer datasets are provided in Table 4.2 and Table 4.3 respectively. The tables contain the name of the targeted miRNA, the references that describe the relevance of the miRNAs in breast cancer or ovarian cancer, the number of mRNAs in the neighborhood network, Pearson correlation coefficient (CC) between mRNA and ground truth spectral count, and lastly, Pearson correlation coefficient between our estimated protein expression and the ground truth spectral count. From the results, we can see that in most cases (i.e., 26 out of 29 in breast cancer and 20 out of 24 in ovarian cancer) the estimated protein expression by considering miRNA regulation achieved a higher correlation with the real protein expression than when only considering mRNA expression.

4.3.2.3 PTNet improved cancer outcome prediction

To provide an additional evaluation of the quality of the estimated protein expression, we designed two cancer outcome prediction tasks by the assumptions that (1) protein expression is a more direct mediator of cellular properties and it will provide more predictive power compared to mRNA expression; (2) a better estimation of protein expression can provide better molecular signatures for cancer outcome prediction. In this experiment, the complete miRNA-mRNA interaction network from TargetScanHuman was applied to estimate the protein expression in the PTNet. The discrim-

Table 4.2: Protein abundance measured by proteomic data to evaluate the accuracy of estimated protein expression in breast cancer dataset. The five columns in the table show the name of the miRNA, the reference of the breast cancer study related to the miRNA, the number of the connected mRNA, correlation coefficients (CC) between the real protein expression and the mRNA expression, and the CC between the real protein expression and the estimated protein expression.

miRNA name	literature	# of connected mRNA	CC of mRNA	CC of protein
hsa-miR-487b	[110]	15	0.305	0.612
hsa-miR-423-3p	[111]	14	0.675	0.798
hsa-miR-10b	[112]	320	0.295	0.403
hsa-miR-506-3p	[113]	1262	0.285	0.314
hsa-miR-1249	[114]	13	0.753	0.859
hsa-miR-296-3p	[115]	70	0.221	0.311
hsa-miR-431	[116]	152	0.286	0.375
hsa-miR-1224-5p	[117]	197	0.348	0.372
hsa-miR-191	[118]	59	0.231	0.309
hsa-miR-376b	[119]	243	0.398	0.469
hsa-miR-324-5p	[120]	142	0.272	0.341
hsa-miR-145	[121]	849	0.254	0.322
hsa-miR-127-3p	[122]	22	0.615	0.675
hsa-miR-154	[123]	162	0.325	0.38
hsa-miR-423-5p	[111]	209	0.351	0.403
hsa-miR-451	[124]	28	0.446	0.498
hsa-miR-802	[125]	362	0.299	0.35
hsa-miR-140-5p	[126]	419	0.301	0.35
hsa-miR-21	[127]	363	0.315	0.329
hsa-miR-29b	[128]	1193	0.289	0.304
hsa-miR-155	[129]	529	0.256	0.271
hsa-miR-125b	[130]	879	0.214	0.254
hsa-miR-221	[131]	480	0.348	0.372
hsa-miR-143-3p	[132]	460	0.283	0.316
hsa-miR-196b	[133]	355	0.457	0.429
hsa-miR-190	[134]	212	0.383	0.364
hsa-miR-146	[135]	270	0.245	0.229

inative power of the estimated protein abundance was compared with mRNA expression and the integration of mRNA and miRNA expressions in the tasks. In each task, the dataset was divided into five folds with three folds for training, one fold for validation (parameter tuning), and one fold for test. A fully connected deep neural network (equation (4.4)) described in the Method Section was applied as the classifier for the estimated protein expression and mRNA expression datasets.

Table 4.3: Protein abundance measured by proteomic data to evaluate the accuracy of estimated protein expression in ovarian cancer dataset. The five columns in the table show the name of the miRNA, the reference of the ovarian cancer study related to the miRNA, the number of the connected mRNA, correlation coefficients (CC) between the real protein expression and the mRNA expression, and the CC between the real protein expression and the estimated protein expression.

miRNA name	literature	# of connected mRNA	CC of mRNA	CC of protein
hsa-miR-487b	[136]	15	0.172	0.545
hsa-miR-423-3p	[137]	14	0.355	0.678
hsa-miR-1249	[114]	13	0.667	0.857
hsa-miR-184	[138]	25	0.352	0.455
hsa-miR-324-5p	[139]	142	0.292	0.39
hsa-miR-10b	[140]	320	0.354	0.439
hsa-miR-329	[141]	338	0.251	0.333
hsa-miR-362-3p	[142]	338	0.251	0.333
hsa-miR-1197	[143]	239	0.29	0.359
hsa-miR-138	[144]	660	0.268	0.334
hsa-miR-502-3p	[145]	196	0.283	0.348
hsa-miR-382	[146]	206	0.27	0.33
hsa-miR-107	[147]	783	0.264	0.324
hsa-miR-145	[148]	849	0.318	0.376
hsa-miR-21	[149]	363	0.326	0.341
hsa-miR-221	[150]	480	0.295	0.311
hsa-miR-29b	[151]	1193	0.260	0.309
hsa-miR-200c	[152]	1144	0.312	0.366
hsa-miR-191	[153]	59	0.484	0.461
hsa-miR-152	[154]	759	0.277	0.265
hsa-miR-1251	[155]	104	0.279	0.270
hsa-miR-328	[156]	193	0.320	0.310

The proposed deep learning-based fusion network (equation (4.6)) was applied to integrate mRNA and miRNA expressions as another baseline for comparison. We repeated the five-fold splitting 100 times by each method on each dataset.

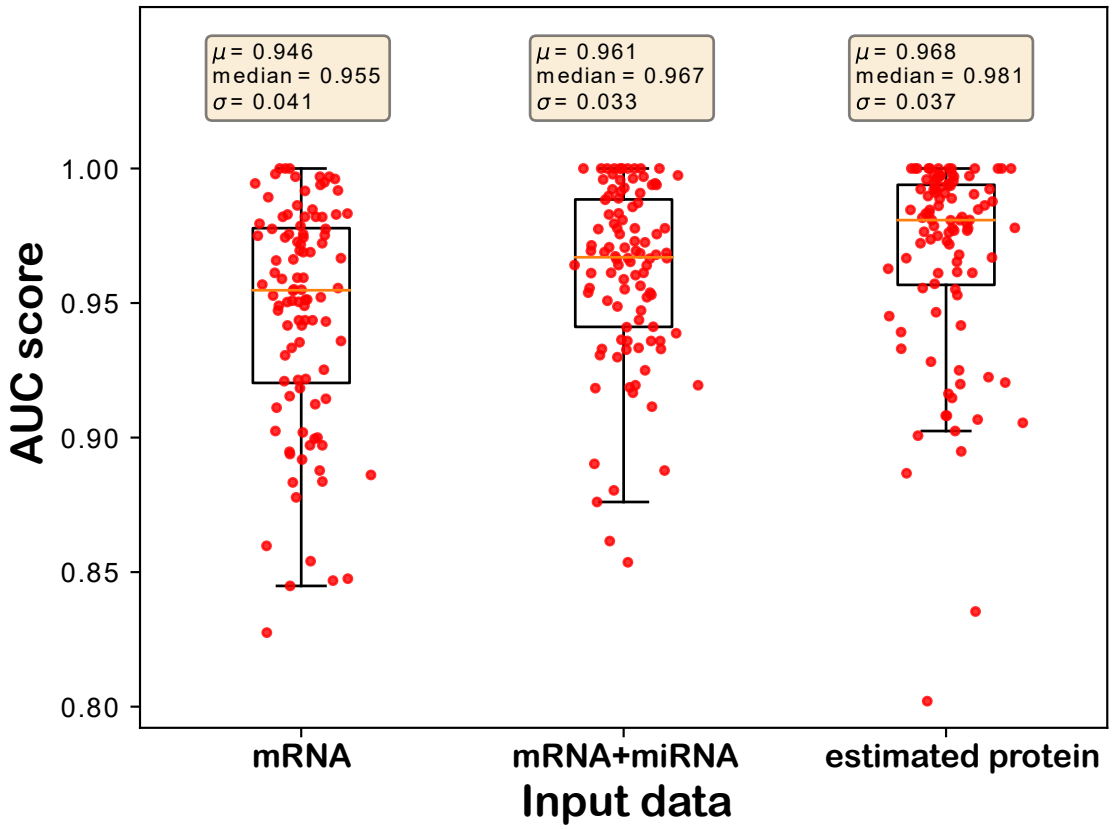

Figure 4.4: Prediction results of the ER status on TCGA breast cancer patients. Each dot represents the AUC score from one splitting. Statistics (mean, median, and standard deviation) of the prediction performance of the 100 splittings are shown above each boxplot.

4.3.2.3.1 Breast cancer

The average area under the curve (AUC) of receiver operating characteristic of the 100 repeats for predicting the ER status of the breast cancer patients are reported in Figure 4.4. Each dot on the boxplot represents the results from one random splitting. Statistics (mean, median, and standard deviation) of the prediction performance of the splitting are shown above each boxplot. The protein expression estimated by PTNet achieved better average classification results (0.968)

Table 4.4: The classification performance on TCGA breast cancer dataset. Average AUC scores and the number of times of win/tie/loss on classification performance between estimated protein expression and the baselines (i.e., mRNA expression and integration of mRNA and miRNA expressions) on breast cancer dataset.

Input data	AUC score	win/tie/loss
mRNA	0.946	11/5/84
mRNA+miRNA	0.961	30/10/60
estimated protein expression	0.968	-

than the ones using mRNA expression (0.946) and the integration of mRNA and miRNA expression (0.961). Since the miRNA expression provides additional predictive signals for breast cancer outcome prediction on top of the mRNA expression, the integration of both with the deep learning-based fusion network model improved the prediction performance compared to the use of mRNA expression only. However, the fusion network model does not consider the miRNA regulation mechanism in its formulation and the classification result is worse than the one using estimated protein expression. In Table 4.4, we also report the number of wins, ties, and losses. The classification results using the mRNA expression and the combination of mRNA and miRNA expression are compared with the results using estimated protein expression. Out of the 100 splittings, the mRNA expression-based prediction only has 11 better predictions than the estimated protein expression whereas the estimated protein expression does a better prediction in 84 splittings. The combination of miRNA and mRNA expressions yields a better prediction than considering the mRNA expression only. The model combining miRNA and mRNA expressions wins 30 splittings against the estimated protein expression but loses in 60 splittings. The overall result shows the consistent improvement of the prediction in breast cancer clinical variables using the estimated protein abundance.

4.3.2.3.2 *Ovarian cancer*

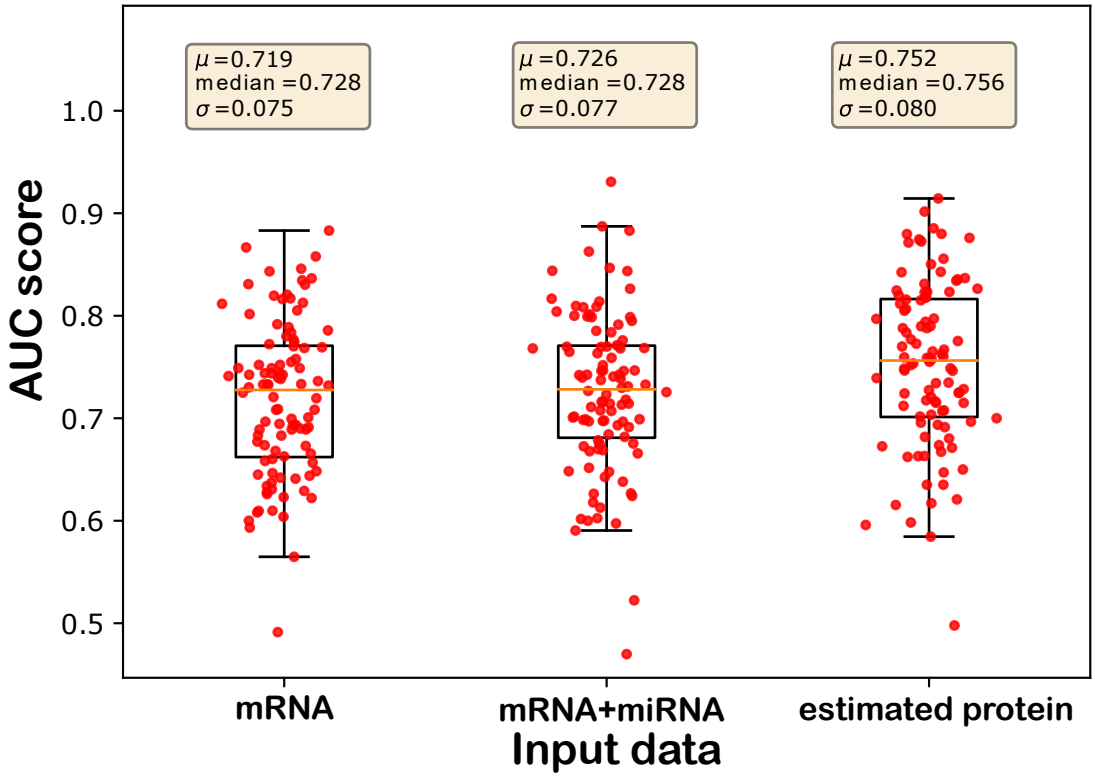

Figure 4.5: Prediction result of cancer stage on TCGA ovarian cancer patients. Each dot represents the AUC score from one splitting. Statistics (mean, median, and standard deviation) of the prediction performance of the 100 splittings are shown above each boxplot.

The results for cancer stage prediction on ovarian cancer patients are illustrated in Figure 4.5. The results show the same trend as on the breast cancer dataset (Figure 4.4), though the overall AUC score is lower than the prediction for the ER status in breast cancer patients. Prediction using the estimated protein expression gives the best AUC score (0.752) followed by the combination of mRNA and miRNA expression (0.726) and mRNA expression (0.719) respectively. Numbers of wins, ties, and losses are also reported in Table 4.5. The superior discriminative power of the estimated protein expression over the mRNA expression and the combination of mRNA and miRNA

expression for both breast cancer and ovarian cancer is illustrated in this section. Therefore, estimated protein expression from PTNet is a more accurate predictor of both breast cancer and ovarian cancer phenotypes compared to mRNA expression and concatenated mRNA and miRNA expression. The improvement in cancer outcome prediction can be attributed to the miRNA-mediated regulation mechanism which we combined with the mRNA expression.

Table 4.5: The classification performance on TCGA ovarian cancer dataset. Average AUC scores and the number of times of win/tie/loss on classification performance between estimated protein expression and the baselines (i.e., mRNA expression and integration of mRNA and miRNA expressions) on ovarian cancer dataset.

Input data	AUC score	win/tie/loss
mRNA	0.719	31/1/68
mRNA+miRNA	0.726	40/0/60
estimated protein expression	0.752	-

4.3.2.4 Effects of APA events

In this subsection, we explored the effects of 3'-UTR APA on miRNA-mediated regulation in two folds: 1) how it changes the miRNA-mRNA interaction; 2) whether a loss of a sponging mRNA due to APA events reroutes miRNAs to other mRNAs and consequently regulate their expression.

To investigate the effects of 3'-UTR APA on the miRNA-mRNA interaction network, the breast cancer patients were divided into the two groups, ER positive and ER negative, and then two lists of mRNAs undergoing APA events corresponding to each group were identified using pipeline APA-Scan [157] which takes aligned bam file for each sample as input. APA-Scan reports the accurate 3'-UTR cleavage site for each mRNA transcript. If the identified cleavage site is upstream of the miRNA-binding position, the transcript will avoid miRNA-mediated regulation and there will be no interactions between the miRNA and the transcript in the network. This process perfectly

illustrates the functional relation between the miRNA-mediated gene regulation and 3'-UTR APA.

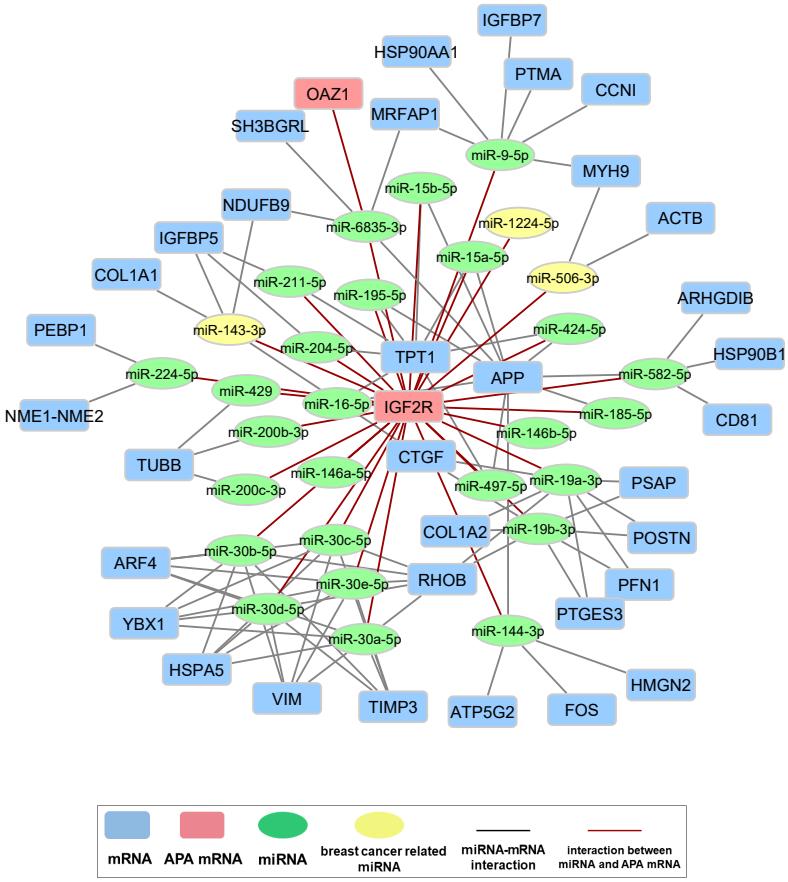

Figure 4.6: miRNA-mRNA interaction network for breast cancer ER positive samples.

In the experiment, we randomly picked a target mRNA from either list of genes undergoing APA events in the ER positive samples or ER negative samples. A sub-network of its neighborhood was built from the complete mRNA-miRNA interaction network. The neighborhood network was defined by a targeted mRNA, all miRNAs directly bind to it (first-order neighbor of targeted mRNA), and mRNAs directly connected (second-order neighbor of targeted mRNA) to the first-order neighbor miRNAs. For presentation purposes, only 40 mRNAs that contain the highest number of interactions with the first-order neighbor miRNAs were selected as the second-order neighbor mRNAs.

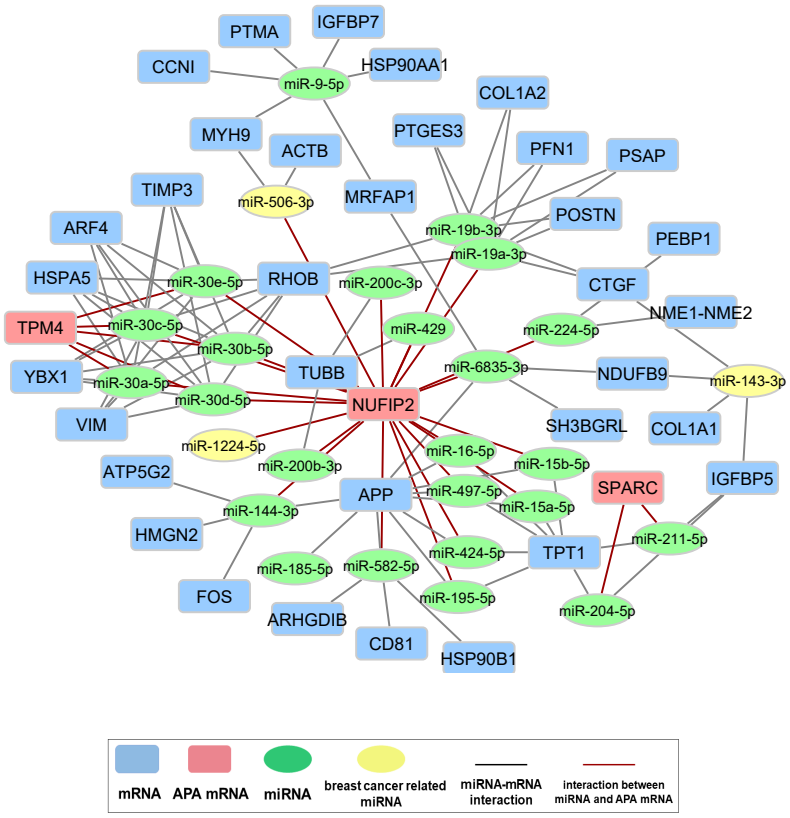

Figure 4.7: miRNA-mRNA interaction network for breast cancer ER negative samples.

Interaction between the selected miRNAs and mRNAs followed the original interaction network. One network for each group (the ER positive or negative group) was then constructed from this sub-network. Figures 4.6 and 4.7 illustrate the two networks for the ER positive and ER negative samples which were built from the same sub-network based on gene *IGF2R* by APA events. In this analysis, we crosschecked the mRNAs present in the sub-network with the list of mRNAs undergoing APA events for the ER positive and ER negative samples respectively. The mRNAs showing APA events in the ER positive samples were deleted from the network along with their miRNA interactions in Figure 4.6. On the other hand, the mRNAs showing APA events in the ER negative samples were marked by red color to indicate that it is only present in the ER positive network

(Figure 4.6). The network for ER negative samples illustrated in Figure 4.7 was constructed in a similar procedure.

As mentioned above, the two networks for the ER positive (Figure 4.6) and ER negative (Figure 4.7) samples were generated from the same sub-network; therefore, they represent the same neighborhood with the exception of the connections with the mRNA undergoing 3'-UTR APA events. The mRNAs showing APA events are marked with red rectangles whereas all other mRNAs are marked with blue rectangles. Thus, mRNAs marked with red rectangles in Figure 4.6 will not exist in Figure 4.7 and vice-versa. All miRNA connections are denoted by gray lines except the mRNAs with APA events, which are marked with red lines. miRNAs are marked by green ovals. Three miRNAs, *miR-506-3p*, *miR-143-3p*, and *miR-1224-5p* that are marked as yellow were listed in Table 4.2 and found as molecular signatures in breast cancer studies. All other connections stay the same between the two networks. These two networks illustrate the dynamic nature of miRNA-mRNA interaction from sample to sample. For example, *IGF2R* marked by the red rectangle in Figure 4.6 is an mRNA undergoing APA in the ER negative samples. Therefore, this mRNA is present in the network of ER positive samples but absent in the network of ER negative samples. In the ER negative samples, this absence causes additional miRNAs to be available for binding to other mRNAs and provides negative regulation of their expression. The prognostic power of APA events in these genes in cancer are well documented in prior studies [158, 159, 160, 161].

In Figure 4.6, *miR-143-3p*, which was shown to play a role in the inhibition of tumor cell proliferation and invasion (Table 4.2), is connected to genes *IGF2R* and *IGFBP5*. *IGF2R* undergoes 3'-UTR APA in the ER negative samples and consequently loses its binding to *miR-143-3p*. As a result, more *miR-143-3p* is available for regulatory binding to *IGFBP5* mRNA. To investigate whether this loss of connection negatively regulates the expression of *IGFBP5* by allowing more *miR-143-3p* to bind to *IGFBP5* mRNA, the changes in the rank of the magnitude of *IGFBP5* expression in the ER positive samples were compared to the ER negative samples. All ranks are

calculated in a descending order of expression. First, in the ER positive samples, changes in the rank ($\Delta R_p = R_{pg} - R_{pp}$) between *IGFBP5* expression among all genes (R_{pg}) and the corresponding protein among all proteins (R_{pp}) is calculated. Then the same approach was taken to calculate the rank for the ER negative samples ($\Delta R_n = R_{ng} - R_{np}$) and compared with each other. We found the change of rank in the ER negative samples to be higher than the change in the ER positive samples (-1955 vs. -1344) (ΔR_n vs. ΔR_p) which signifies the negative regulatory effect of miRNA on the *IGFBP5* expression. The ranking comparison can be interpreted in such a way that the drop of *IGFBP5* expression ranking in the proteome of ER negative samples is higher than that of ER positive samples. *IGF2R* mRNA, on the other hand, being free from miRNA inhibition rose its ranking higher in the proteome of ER negative samples than the ER positive ones (902 vs. 426) (ΔR_n vs. ΔR_p). Therefore, this experiment demonstrates how 3'-UTR APA events change the miRNA-mRNA interaction(s) and cause negative regulation on the expression of mRNAs.

4.4 Discussion

Although the proteome mostly determines biology and clinical outcomes in human disease pathogenesis, the application of current proteome profiling technologies is less exhaustive than transcriptome profiling due to technical limitations such as the dynamic range of data acquisition. Thus, transcriptome profiling using RNA-seq experiments is widely used instead to understand the gene expression in most big data-driven studies. Despite such popularity, the data analysis has been one-dimensional in such a way that differential gene expression analysis has been a standard procedure for most data processing. It limited a comprehensive understanding of the role of the transcriptome by excluding the post-transcriptional regulations and incurred a pervasive problem of poor correlation between the transcriptome and the proteome in big data-driven studies. In this study, we argue that PTNet, a multi-dimensional data analysis model, can overcome the problems

in current data analyses and provide evidence that it performs better in assessing the proteome changes and improves the prediction of clinical outcomes compared to current data analysis tools.

Our model highly considers the changes of miRNA-binding sites in the transcriptome. Previously, it was suggested that ceRNAs can modulate the regulatory mechanism of miRNAs [162, 163]. However, in this model, the expression level of ceRNAs has been the major focus as miRNAs were known to target multiple mRNAs in cells. As miRNAs are known to bind to 3'-UTR of mRNAs for the regulation of gene expression, the qualitative and quantitative information on 3'-UTR APA events is critical to understand the regulatory network of miRNAs. So far, numerous bioinformatics pipelines for 3'-UTR APA events have been developed using RNA-seq or 3'-end biased RNA-seq [164, 157, 165, 166]. Although they provide a comprehensive profile of 3'-UTR APA events, we demonstrated that integrating two sequencing results (RNA-seq and 3'-end biased RNA-seq) could provide a better resolution of 3'-UTR APA profiling [165]. In this regard, it would be important to develop pipelines that could provide a higher resolution of 3'-UTR APA profiling by considering various RNA-seq resources.

4.5 Summary

In this study, we introduce a graph-based learning model to predict protein expression in cells. Our model focuses on two particular post-transcriptional regulatory mechanisms in gene expression; miRNA-mediated gene regulation and 3'-UTR APA events. A deep learning-based fusion network was also proposed to combine the mRNA and miRNA expression profiles without considering the miRNA-mRNA interactions as a baseline method. We observed the estimated protein expression is more consistent with the true protein expression and has more discriminative power to classify clinical variables of cancer patients compared to either the mRNA expression or the combination of mRNA and miRNA expression. We also analyzed the effect of 3'-UTR APA events on the

competing endogenous RNA model where multiple targeting capacity of miRNAs can show the dynamic relationship with their target mRNAs with an intuition that an mRNA losing its miRNA-binding site will result in the regulation of other mRNAs by the same miRNA. Our results show the negative regulation caused by miRNA when one of its neighboring mRNAs undergo 3'-UTR APA. Our findings in this study signify the importance of considering post-transcriptional regulation in cancer research. The proposed efficient and scalable computational methods enable a better understanding of the molecular basis of cancer pathogenesis and provide a previously unrecognized perspective in cancer data mining.

CHAPTER 5: MULTI-MODAL MISSING VALUE IMPUTATION

The work in this chapter has been published in the following paper:

Khandakar Tanvir Ahmed, Sudipto Baul, Yanjie Fu, and Wei Zhang (2023). Attention-Based Multi-modal Missing Value Imputation for Time Series Data with High Missing Rate. In Proceedings of the 2023 SIAM International Conference on Data Mining (SDM) (pp. 469-477). Society for Industrial and Applied Mathematics. [167]

Previous chapters have shown the effectiveness of multi-modal integration for various downstream tasks. The performance of these models depend on the quality and availability of input data. High rate of missing values in the data modalities will adversely affect our ability to meaningfully integrate the datasets. This chapter focuses on techniques for multi-modal data imputation that can in turn improve the performance of integrative models.

5.1 Introduction

Multivariate time series data has an important bearing in many domains such as healthcare [168, 169], finance [170], and meteorology [171]. The ability of time series data to capture changes in the system over time has made it popular in the research community. Many advanced algorithms have been proposed for information extraction and pattern recognition in time series data to perform various downstream tasks [172]. However, time series data is prone to incompleteness due to the prolonged data collection procedures. Data can be missing due to damaged collecting devices, device malfunction, sabotage, or participant not showing up for data collection [173]. Missing

data impedes the use of statistical analysis on the dataset to find meaningful patterns. Therefore, handling missing values in time series data has long been a key challenge for researchers.

Handling techniques of missing values in a time series data can be broadly divided into two classes. The first class is case deletion where incomplete observations are removed from the analysis [28]. This is a useful approach if the missing rate is low. As the missing rate increases, case deletion presents a significant drawback by ignoring important information in deleted data. The second approach is imputing the missing value with a reasonable estimation. It can be simple imputation methods such as mean imputation, median imputation, and last observation imputation. However, these techniques fail to utilize temporal information as well as capture the relation among features of the same observation in the time series data. There are also more advanced machine learning-based algorithms for missing value imputation. e.g. KNN based imputation [29], Matrix Factorization-based imputation [30], and maximum likelihood Expectation-Maximization (EM) based imputation [31]. Although they can capture relations among features, they still cannot exploit temporal information. Recently, deep learning-based imputations, powered by recurrent neural networks, and generative adversarial networks have shown remarkable success in estimating missing values due to their ability to interpret temporal dependency in data and map complex relations among features [32, 33].

Existing studies for time series imputation are uni-modal and self-imputation where the missing values are imputed only using the available values in the same dataset [34, 35]. However, the real world is filled with multi-modal time series data that is being increasingly used in studies [36], thanks to the advancement in data collection and processing technologies. Generally, data from different modalities contain complementary information [37, 38] and the introduction of this complimentary information can further improve the missing value estimation over existing self-imputation models. Multi-modal imputation for cross-sectional data has already shown success [39] which can also be extended to the time series domain. Nonetheless, multi-modal time series

imputation comes with some unique challenges. The first challenge is, one of the data can be cross-sectional which means we need a model that can effectively map cross-sectional data to another time series. The second challenge is that some samples can have no available time series data. This may happen if the cross-sectional data is collected for a larger population compared to the time series data due to expensive and logistically difficult data collection [40]. Multi-modal imputation can help us estimate the data for these completely missing samples which is by default not possible in uni-modal imputation techniques.

The self-attention mechanism [174] has established itself as the primary tool for sequence modeling in recent times. It enables more parallelization and better capture of temporal information compared to recurrent neural networks (RNN)[175], long short-term memory (LSTM)[176], and gated recurrent neural networks (GRU)[177]. Despite being state-of-the-art in many time series domains, the use of self-attention is still limited for missing value imputation in time series data. Lately, few works show the potential of such model to impute missing data in a time series [178, 179]. However, all of these studies are uni-modal and not designed to harness multiple data streams.

5.1.1 Contribution

In this study, we propose a multi-modal time series imputation framework that, TSEst, offers advantages over the existing literature in the following ways. 1) It can integrate an additional stream of information from another data modality for a better estimation of missing values. 2) The model can efficiently map cross-sectional data to time series, thus reducing reliance on missing value-prone time series data. 3) Samples with completely missing data can also be imputed using this framework due to the presence of an extra data modality. A comprehensive set of experiments on two datasets show improved performance of our proposed model over the state-of-the-art baselines.

5.2 Problem Statement

Let $\boldsymbol{X}$ and $\boldsymbol{Y}$ be two datasets collected to describe the same participant/sample. $\boldsymbol{X}$ is an incomplete time series data with missing values that need to be imputed whereas $\boldsymbol{Y}$ is a complete cross-sectional data with no missing values. Our objective is to impute the missing values in $\boldsymbol{X}$ with reasonable estimations using available values in $\boldsymbol{X}$ and data from $\boldsymbol{Y}$. One incomplete time series data $\boldsymbol{X}$ with p features, observed in $T = (1, 2, ..., t)$ is defined as $\boldsymbol{X} = (\boldsymbol{x}_1, ..., \boldsymbol{x}_i, ...\boldsymbol{x}_t) \in \mathbb{R}^{t \times p}$ where $\boldsymbol{x}_i$ denotes the observation of $\boldsymbol{X}$ in i^{th} time step and $\boldsymbol{x}_i^j$ is the observation of j^{th} features in i^{th} time step. Mask matrix $\boldsymbol{M} \in \mathbb{R}^{t \times p}$ is introduced to keep track of missing values in $\boldsymbol{X}$:

$$M_i^j = \begin{cases} 1, & \text{if } \boldsymbol{x}_i^j \text{is observed} \\ 0, & \text{otherwise} \end{cases}$$

A fixed percentage of values in $\boldsymbol{X}$ are removed by artificial masking to obtain $\tilde{\boldsymbol{X}}$. $\tilde{\boldsymbol{X}}$ is used as input to the model whereas $\boldsymbol{X}$ works as the ground truth to train and evaluate the model. Let $\boldsymbol{M}_1$ be the mask matrix indicating the artificially masked values. The other dataset, $\boldsymbol{Y}$ is a cross-sectional data defined as $\boldsymbol{Y} \in \mathbb{R}^{1 \times q}$ where q is the number of features. Before introducing the proposed framework TSEst, we discuss the challenges in the domain of time series imputation in this section along with the novelty and potential of our model to solve them. Different challenges in time series imputation can be categorized as follows:

(1) **Uni-modal vs Multi-modal imputation** Depending on the availability of data, imputation can be uni-modal [173] where missing values in $\boldsymbol{X}$ are imputed using only the available data in $\boldsymbol{X}$. It can also be multi-modal if more than one dataset is available for the same set of samples such as [180, 181] where they collected multiple time series and cross-sectional data from the same cohort. In multi-modal imputation, we impute missing values in $\boldsymbol{X}$ with the help of $\boldsymbol{Y}$ and available data

in $\boldsymbol{X}$.

(2) **Partial vs Complete missing** There are two possibilities for the amount of missing data. Some variables or time steps in $\boldsymbol{X}$ can be missing creating a partially missing dataset. Alternatively, all time steps can be missing i.e. the time series data $\boldsymbol{X}$ was never collected for a participant, referred to as completely missing time series in this study.

Moreover, there are two likely missing patterns in a partially missing time series data. First, variables are randomly missing in a time step because the values were not captured, or the captured values were corrupted. Second, variables are missing in a chunk when the data collecting procedure is interrupted for an extended period of time. This is a common scenario in biomedical studies as the data is often collected from living persons. If the participant skips a visit then all variables in that time step are missing. The followings are examples of random and chunk missing values with $t = 4$ and $p = 3$ and "none" representing missing values. (a) shows random missing values and (b) shows chunk missing values with $t = 3$ missing. A feature such as $p = 3$ missing for an extended amount of time is also considered as chunk missing.

$$\boldsymbol{X} = \begin{bmatrix} 10 & none & 6 \\ 3 & 14 & none \\ none & 5 & 8 \\ 9 & none & none \end{bmatrix}$$

(a) Missing in random

$$\boldsymbol{X} = \begin{bmatrix} 10 & 4 & 6 \\ 3 & 14 & 10 \\ none & none & none \\ 9 & 6 & 5 \end{bmatrix}$$

(b) Missing in chunk

(3) **Time series-time series vs Cross sectional-time series imputation** Multi-modal imputations can further be divided into two classes as the data $\boldsymbol{Y}$ available to estimate missing values in $\boldsymbol{X}$ can either be a time series or cross-sectional data. For example, [180] collects two time series data from the samples where one dataset can be used to impute missing values in the other one. On the contrary, [40] collects a cross-sectional and a time series data from its participants. In this case, the cross-sectional data will be used to impute missing values in the time series to contribute

to the imputation. There are several advantages of cross-sectional to time series imputation over time series-time series imputation. First, limited by cost and logistics, time series data is often collected for a subset of the participants in cross-sectional data [40]. Therefore, an effective cross-sectional to time series imputation model can help reduce the reliance on an expensive and long-term longitudinal data collection and still provide us with reasonable data estimation for a large number of samples. Additionally, if a time series data has missing values, it is a fair assumption that the other time series data collected from the same sample set will be incomplete as well. So, it is imperative to develop a model that can estimate missing values in a time series using cross-sectional data.

5.3 TSEst Imputation Framework

The framework TSEst consists of three components as illustrated in Fig 5.1. $\mathbf{Component\ 1}$ takes the cross-sectional data $\boldsymbol{Y}$ as input and extracts meaningful information to generate an intermediate time series data $\boldsymbol{Z}_1$. $\mathbf{Component\ 2}$ takes the time series $\tilde{\boldsymbol{X}}$ as input and generates another intermediate time series data $\boldsymbol{Z}_2$. Both these time series data are merged into a single data using a weighted addition technique and fed into $\mathbf{Component\ 3}$ that makes the final prediction for the missing values. The notations used in this manuscript are summarized in Table 5.1.

5.3.1 Overview of the workflow

Component 1 has two elements: one is a fully connected feed forward neural network (FFNN) and a self-attention block (SA block) that follows the FFNN. $\boldsymbol{Y}$ is used as input to FFNN and transformed into $\boldsymbol{Z}_{FFNN} \in \mathbb{R}^{1\times p}$ following equation 5.1.

$$\boldsymbol{Z}_{out} = \{\sigma(\boldsymbol{W}\boldsymbol{Z}_{in} + \boldsymbol{b})\}^{\mathcal{N}} \tag{5.1}$$

Table 5.1: Notations for the proposed model

Name	Definition
$\boldsymbol{X} \in \mathbb{R}^{t \times p}$	Incomplete time series data
$\tilde{\boldsymbol{X}} \in \mathbb{R}^{t \times p}$	Artificially masked time series data
$\boldsymbol{Y} \in \mathbb{R}^{1 \times q}$	Cross-sectional data
$\boldsymbol{M} \in \mathbb{R}^{t \times p}$	Mask matrix to represent missing values in $\boldsymbol{X}$
$\boldsymbol{M}_1 \in \mathbb{R}^{t \times p}$	Mask matrix indicating artificially missing values in $\boldsymbol{X}$
$\boldsymbol{Z}_{FFNN} \in \mathbb{R}^{1 \times p}$	Output from the feed forward neural network (FFNN)
$\boldsymbol{Z}_1 \in \mathbb{R}^{t \times p}$	Output from the SA block in Component 1
$\boldsymbol{Z}_2 \in \mathbb{R}^{t \times p}$	Output from the SA block in Component 2
$\boldsymbol{Z}_3 \in \mathbb{R}^{t \times p}$	Weighted addition of $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$
$\hat{\boldsymbol{X}} \in \mathbb{R}^{t \times p}$	Predicted data for $\boldsymbol{X}$ in Component 3

where σ denotes the activation function and $\boldsymbol{W}$, $\boldsymbol{b}$ are the learnable parameters. $\mathcal{N}$ stands for the number of stacked layers. $\boldsymbol{Z}_{in}$ for the first layer is $\boldsymbol{Y}$ and $\boldsymbol{Z}_{out}$ in the last layer is $\boldsymbol{Z}_{FFNN}$. $\boldsymbol{Z}_{FFNN}$ is replicated t times and fed into SA block. Each SA block uses self-attention mechanism to capture or create time dependency in the input data to generate a plausible synthetic time series as output. Details of SA block functionality will be provided in section 5.3.2. The output from the SA block in Component 1, given by $\boldsymbol{Z}_1 \in \mathbb{R}^{t \times p}$ represents a complete approximation for $\boldsymbol{X}$ from $\boldsymbol{Y}$. It should be noted that the SA block in Component 1 does not extract any temporal information from its input $\boldsymbol{Z}_{FFNN}$ as it is not a time series data. $\tilde{\boldsymbol{X}}$ is used in the training of Component 1 to induce temporal factor in $\boldsymbol{Z}_1$ and learn how $\boldsymbol{Y}$ relates to that temporal element. In another word, the cross-sectional data can have different impact on different time steps of the time series data which we aim to model in the estimation using Component 1.

Component 2 consists of another SA block that takes the concatenation of $\tilde{\boldsymbol{X}}$ and $\boldsymbol{M}$ as input. The mask matrix $\boldsymbol{M}$ helps the model to learn which values are missing in $\tilde{\boldsymbol{X}}$ and should be ignored. Output from Component 2, $\boldsymbol{Z}_2 \in \mathbb{R}^{t \times p}$ is another complete approximation for $\boldsymbol{X}$. $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$ are merged into a single dataset using a weighted addition block, which will be described in section

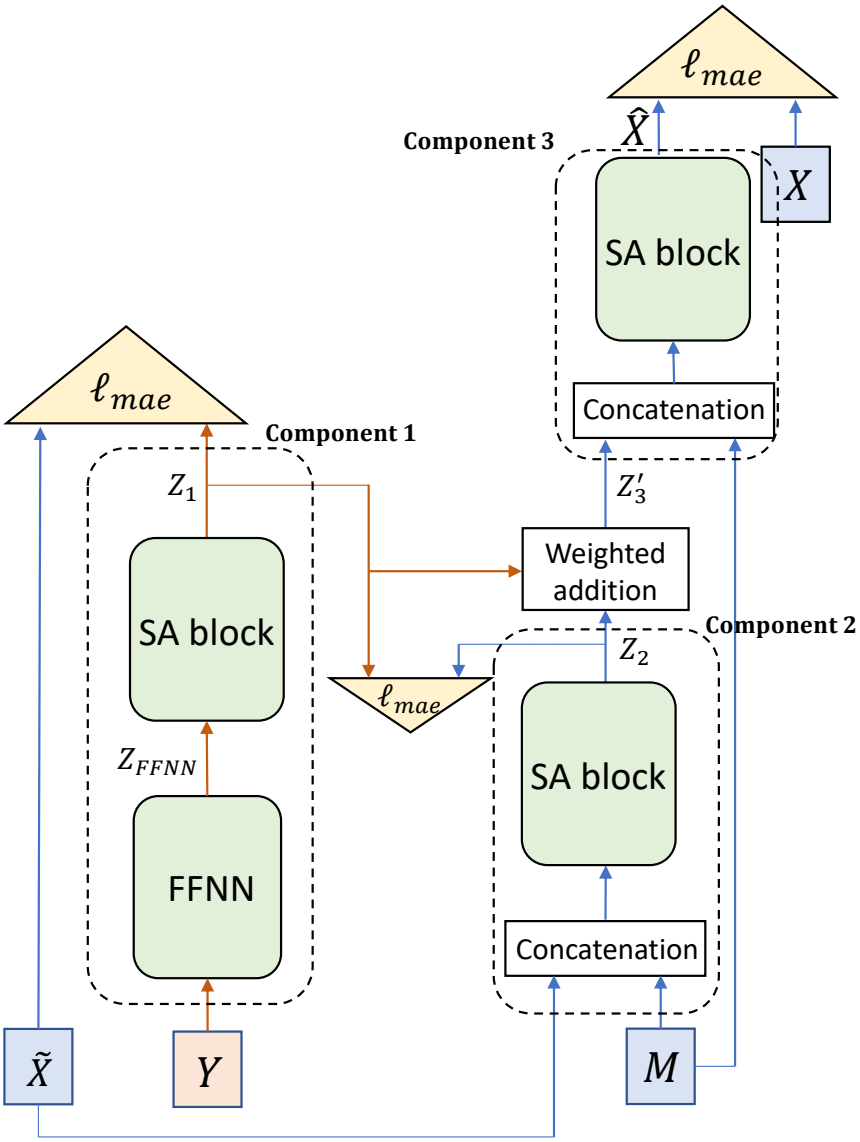

Figure 5.1: Overview of the proposed framework TSEst.

5.3.2. Each time step in $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$ are checked for their accuracy against available data in $\tilde{\boldsymbol{X}}$. Then we find which time steps contributed more to those accurate estimations to calculate their weights. Once we have calculated weights $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ for $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$ respectively, we merge them into $\boldsymbol{Z}_3 \in \mathbb{R}^{t \times p}$ following equation 5.1, where $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are the weights for $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$ respectively.

$$\boldsymbol{Z}_3 = \boldsymbol{\alpha}^T * \boldsymbol{Z}_1 + \boldsymbol{\beta}^T * \boldsymbol{Z}_2 \tag{5.2}$$

$\boldsymbol{Z}_3$ is a combination of synthetic time series data generated from both $\tilde{\boldsymbol{X}}$ and $\boldsymbol{Y}$. Values already present in $\tilde{\boldsymbol{X}}$ are used to replace the synthetic values in $\boldsymbol{Z}_3$ following equation 5.3, where $\odot$ is the Hadamard product.

$$\boldsymbol{Z}_3^{'} = \boldsymbol{M} \odot \tilde{\boldsymbol{X}} + (1 - \boldsymbol{M}) \odot \boldsymbol{Z}_3 \tag{5.3}$$

Component 3 is also comprised of an SA block that uses $\boldsymbol{Z}_3^{'}$ concatenated with $\boldsymbol{M}$ as input to generate $\hat{\boldsymbol{X}}$. $\hat{\boldsymbol{X}}$ is the final output from the imputation model. If $\boldsymbol{Y}$ is a time series data as well, we use the same model, except a long short-term memory (LSTM) in place of FFNN in Component 1. Although the model is designed to be multi-modal, in case of data unavailability, it can be used as uni-modal by turning off Component 1.

5.3.2 Proposed modules

5.3.2.1 SA block

Self-attention (SA) block is used to capture or create time dependency in a given time series input. We employ the self-attention mechanism proposed by [174] to build the SA blocks. For a time series $\boldsymbol{X}_a \in \mathbb{R}^{t \times p_a}$, where p_a is the number of input features, it is first embedded into a new feature space of size p_e. Positional encoding $\boldsymbol{P}$ is added with the new feature space to produce $\boldsymbol{X}_e \in \mathbb{R}^{t \times p_e}$ following equation 5.4.

$$\boldsymbol{X}_e = [\boldsymbol{X}_a \boldsymbol{W}_e + \boldsymbol{b}_e] + \boldsymbol{P} \tag{5.4}$$

We adopt the sine and cosine functions of different frequencies to represent positional encoding $\boldsymbol{P}$:

$$\boldsymbol{P}_{(pos,2i)} = \sin(pos/10000^{2i/p_e})$$

$$\boldsymbol{P}_{(pos,2i+1)} = \cos(pos/10000^{2i/p_e})$$

where pos is the time step position and i is the dimension along p_e. $\boldsymbol{X}_e$ is then linearly mapped into query ($\boldsymbol{Q}$), key ($\boldsymbol{K}$), and value ($\boldsymbol{V}$) of dimension d_k, d_k, and d_v respectively:

$$\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V} = \boldsymbol{X}_e\boldsymbol{W}_Q, \boldsymbol{X}_e\boldsymbol{W}_K, \boldsymbol{X}_e\boldsymbol{W}_V$$

where $\boldsymbol{W}_Q \in \mathbb{R}^{p_e \times d_k}$, $\boldsymbol{W}_K \in \mathbb{R}^{p_e \times d_k}$ and $\boldsymbol{W}_V \in \mathbb{R}^{p_e \times d_v}$ are the learnable parameters. The product between $\boldsymbol{Q}$ and $\boldsymbol{K}$ is scaled using d_k, dimension of the key vector, to obtain the attention scores in equation 5.5 and passed through *softmax* to get the attention weights. The output is computed as follows:

$$\boldsymbol{H} = \textit{SelfAttention}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \textit{Softmax}(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d_k}})\boldsymbol{V} \tag{5.5}$$

This architecture of finding the output is called a head and we denote the output from each head as $\boldsymbol{H} \in \mathbb{R}^{t \times d_v}$. We set the diagonal elements in the attention matrix to zero to make the model robust by preventing a time step from contributing to their own estimation [179]. Multi-head attention scheme is employed to stabilize the training and capture a broader range of relationships between time steps. In multi-head attention with h heads, the output is computed h times with different learned parameters and concatenated to obtain a single output following equation 5.6.

$$\bar{\boldsymbol{X}}_e = \textit{MultiHead}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = (\|_{k=1}^{h} \boldsymbol{H}_k)\boldsymbol{W}_o \tag{5.6}$$

where $\boldsymbol{H}_k$ is the output from k^{th} head and $\boldsymbol{W}_o \in \mathbb{R}^{hd_v \times p_e}$ is learnable parameter. Let the output of the multi-head attention is $\bar{\boldsymbol{X}}_e$. A position-wise feed-forward network (equation 5.7)is applied to

the output $\bar{\boldsymbol{X}}_e$.

$$PFF(\bar{\boldsymbol{X}}_e) = ReLU(\bar{\boldsymbol{X}}_e \boldsymbol{W}_1 + \boldsymbol{b}_1)\boldsymbol{W}_2 + \boldsymbol{b}_2 \tag{5.7}$$

where $\boldsymbol{W}_1 \in \mathbb{R}^{p_e \times d_f}$, $\boldsymbol{b}_1 \in \mathbb{R}^{d_f}$, $\boldsymbol{W}_2 \in \mathbb{R}^{d_f \times p_e}$, $\boldsymbol{b}_2 \in \mathbb{R}^{p_e}$. Finally, $PFF(\bar{\boldsymbol{X}}_e)$ is projected back to the original feature space of input data and stands as the output from the SA block.

5.3.2.2 Weighted addition

Two synthetic time series data generated by two SA blocks are merged into a single time series in the proposed weighted addition block by taking the attention weights in their respective SA blocks into consideration. $t \times t$ attention weight matrices are extracted from each head of the two SA blocks that generate synthetic time series data $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ respectively. Attention weights from all heads are averaged into $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ corresponding to the attentions for $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ respectively. i^{th} row in the attention matrix tells us the contribution of each time step in the generation of synthetic data at i^{th} time step. Our objective is to identify the time steps that contribute more towards an accurate estimation of synthetic data and assign a higher weight to those time steps during the addition.

Let $\boldsymbol{s}_i^1$ and $\boldsymbol{s}_i^2$ be the i^{th} row/time step in the $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ respectively. Similarly, $\boldsymbol{a}_i^1$ and $\boldsymbol{a}_i^2$ denote the i^{th} row in the attention weight matrices $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ respectively. The correlations c_1 and c_2 between true data $\boldsymbol{x}_i$ and two synthetic data $\boldsymbol{s}_i^1$ and $\boldsymbol{s}_i^2$ at time step i are calculated and modified according to lines 2-4 in Algorithm 1, where τ and κ are parameters to adjust the decaying function. We used $\tau = 3$ and $\kappa = 5$ in this study. c_1 and c_2 are then normalized using lines 5-6 to find the relative accuracy of $\boldsymbol{s}_i^1$ and $\boldsymbol{s}_i^2$ in estimating $\boldsymbol{x}_i$. We multiply the normalized correlation values c_1 and c_2 with $\boldsymbol{a}_i^1$ and $\boldsymbol{a}_i^2$ respectively to obtain $\boldsymbol{A}_1^{'}$ and $\boldsymbol{A}_2^{'}$. $\boldsymbol{A}_1^{'}$ and $\boldsymbol{A}_2^{'}$ are averaged along the rows (column mean) and again normalized following lines 14-15. i^{th} value in $\boldsymbol{\alpha}$ represents the importance of i^{th} time step in generating an accurate estimation of other time steps. i^{th} value in $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are

Algorithm 1 Weighted addition algorithm

Input: $\boldsymbol{A}_1$, $\boldsymbol{A}_2$, $\boldsymbol{S}_1$, $\boldsymbol{S}_2$, and $\boldsymbol{X}$
Output: $\boldsymbol{X}^{'}$

for $i = 1 \rightarrow t$ **do**
  $c_1 \leftarrow$ *correlation*$(\boldsymbol{x}_i, \boldsymbol{s}_i^1)$
  $c_2 \leftarrow$ *correlation*$(\boldsymbol{x}_i, \boldsymbol{s}_i^2)$
  $c = \begin{cases} c, & \text{if } c \geq 0 \\ \frac{\exp^{\tau c}}{\kappa}, & \text{otherwise} \end{cases}$
  $c_1 \leftarrow c_1/(c_1 + c_2)$
  $c_2 \leftarrow c_2/(c_1 + c_2)$
  $\boldsymbol{a}_i^{1'} \leftarrow \boldsymbol{a}_i^1 * c_1$
  $\boldsymbol{a}_i^{2'} \leftarrow \boldsymbol{a}_i^2 * c_2$
end for
$\boldsymbol{A}_1^{'} = (\boldsymbol{a}_i^{1'}, ..., \boldsymbol{a}_i^{1'}, ...\boldsymbol{a}_t^{1'})$
$\boldsymbol{A}_2^{'} = (\boldsymbol{a}_i^{2'}, ..., \boldsymbol{a}_i^{2'}, ...\boldsymbol{a}_t^{2'})$
$\boldsymbol{A}_1^{'} \leftarrow mean\ along\ the\ rows(\boldsymbol{A}_1^{'})$
$\boldsymbol{A}_2^{'} \leftarrow mean\ along\ the\ rows(\boldsymbol{A}_2^{'})$
$\boldsymbol{\alpha} \leftarrow \boldsymbol{A}_1^{'}/(\boldsymbol{A}_1^{'} + \boldsymbol{A}_2^{'})$
$\boldsymbol{\beta} \leftarrow \boldsymbol{A}_2^{'}/(\boldsymbol{A}_1^{'} + \boldsymbol{A}_2^{'})$
$\boldsymbol{X}^{'} = \boldsymbol{\alpha}^T * \boldsymbol{S}_1 + \boldsymbol{\beta}^T * \boldsymbol{S}_2$

multiplied with i^{th} row in $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ to derive a scaled synthetic data. The underlying hypothesis is if the synthetic data is accurate at one time step, then the other time steps that contributed more to its generation must be accurate as well.

5.3.3 Missing value imputation

For an incomplete time series $\tilde{\boldsymbol{X}}$, we estimate a synthetic time series $\hat{\boldsymbol{X}}$ that closely resembles the true values in $\tilde{\boldsymbol{X}}$. The loss function to train the model has three elements, one on each of the component's output. We use both reconstruction and imputation loss for the training. Reconstruction loss is defined as the error in estimating the available values that the model is allowed to see during training computation. In contrast, imputation loss refers to the error in estimating the artificially

masked values. The loss function is given by equation 5.8.

$$\begin{aligned}\mathcal{L} = \ell_{mae}(\boldsymbol{M}, \boldsymbol{X}, \hat{\boldsymbol{X}}) + \lambda \ell_{mae}(\boldsymbol{M}_1, \boldsymbol{X}, \hat{\boldsymbol{X}}) + \\ \gamma \ell_{mae}(\boldsymbol{M}, \tilde{\boldsymbol{X}}, \boldsymbol{Z}_1) + \mu \ell_{mae}(\boldsymbol{Z}_1, \boldsymbol{Z}_2)\end{aligned} \tag{5.8}$$

where λ, γ, and μ are variable weights on different parts of the equation. $\ell_{mae}(\boldsymbol{M}, \boldsymbol{X}, \hat{\boldsymbol{X}})$ calculates the reconstruction loss of observed values in the training data and $\ell_{mae}(\boldsymbol{M}_1, \boldsymbol{X}, \hat{\boldsymbol{X}})$ measure imputation accuracy for the artificially masked values. $\ell_{mae}(\boldsymbol{M}, \boldsymbol{Z}_1, \tilde{\boldsymbol{X}})$ ensures that the output from Component 1 which is generated from a cross-sectional data, learns to imitate the true time series data, therefore, contribute meaningful information for the final estimation. $\ell_{mae}(\boldsymbol{Z}_1, \boldsymbol{Z}_2)$ makes the output from Component 1 and Component 2 similar which is particularly helpful for predicting values for completely missing samples. That way, even if test samples do not have any time series data at all, Component 1 can still make a reasonable prediction similar to the expected output from Component 2. The impact of this element on partially missing samples can be controlled using μ.

5.4 Experiments

We evaluate the performance of our proposed framework TSEst using two datasets. Imputation accuracy of TSEst is compared with state-of-the-art recurrent neural network (RNN)-based methods and attention-based transformer in two different missing value patterns. Moreover, we investigate the ability of our framework to harness information from the cross-sectional data and map that to the time series.

5.4.1 Dataset and tasks

5.4.1.1 TEDDY

TEDDY is a longitudinal cohort study designed to find the factors affecting the progression towards type 1 diabetes (T1D) [182]. They collected various clinical and omics datasets from the enrolled participants including single nucleotide polymorphism (SNP) and gene expression. The time series gene expression contains 401 samples with observations of 17039 features at 16 time steps. 79% time steps in the gene expression are missing. This dataset only contains missing time steps i.e. if a time step is missing, all 17039 features are missing. This missing pattern resembles chunk missing as described previously. On the other hand, cross-sectional SNP data with 176,586 features have no missing values. We reduced the dimension of SNP to 50 using principal component analysis (PCA) to avoid overfitting the framework.

Table 5.2: Dataset statistics

Dataset	samples	length	features	missing rate
TEDDY	401	16	50/17,039	78.63%
Maurer	531	500	27/6	0%
DayMet	531	500	27/14	0%

5.4.1.2 CAMELS

For further evaluation of our model, we used the Catchment Attributes and Meteorological dataset for Large-sample Studies (CAMELS) datasets [183]. We used two time series data from CAMELS, **Maurer** ($\frac{1}{8}th$ degree spatial resolution) and **DayMet** ($1km \times 1km$ spatial resolution), collected for 531 basins following the benchmarking by Newman et al[184]. 500 time steps of 6 and 14 different weather or soil states attributes in Maurer and DayMet datasets respectively were used in this study

along with 27 static features as cross-sectional data. There is no missing value in the datasets. The statistics of TEDDY and CAMELS datasets are presented in Table 5.2. The *features* column reports the number of features in cross-sectional data followed by the number of features in the time series data.

5.4.2 Experimental setup

Time series datasets are artificially masked before imputation to be able to assess the imputation accuracy. The artificial masking is done in two different ways. For $r\%$ random missing, $r\%$ of available values are masked in the dataset randomly whereas chunk missing masks $r\%$ of available time steps entirely. 80% of the samples are used in the training and the rest are used as test set to evaluate the models. Further 20% of the training samples are kept for validation to tune the hyper-parameters and select the best trained model.

5.4.3 Comparison of time series imputations

To evaluate the quality of the imputed data generated by our proposed model TSEst, we designed three experiments on TEDDY and CAMELS datasets under the assumptions: (1) the imputed values learned from our proposed model harness two data modalities, therefore, will generate imputations with higher quality; (2) the impact of the additional cross-sectional data will be more apparent with higher missing rate. The performance is compared against three uni-modal baselines: BRTIS [185], M-RNN [32], and Transformer [174]. For all tasks, different percentage of available data in the time series data is artificially masked and used as ground truth to measure the imputation accuracy. Tables 5.3 and 5.4 shows the imputation accuracy for randomly missing values and chunk missing values in the time series data at 20% artificial missing rate. The results are presented in terms of root means square error (RMSE) and mean absolute error (MAE).

Table 5.3: RMSE/MAE of the imputation on test set [random missing]

Method	TEDDY	Maurer	DayMet
Mean filling	2.533/0.529	0.826/0.602	0.706/0.370
Last filling	2.538/0.557	0.723/0.430	0.672/0.247
M-RNN	—/—	0.763/0.427	0.725/0.467
BRITS	—/—	0.343/0.162	0.443/0.152
Transformer	0.339/0.224	0.232/0.089	0.306/0.096
TSEst	0.294/0.178	0.238/0.088	0.223/0.065

Table 5.4: RMSE/MAE of the imputation on test set [chunk missing]

Method	TEDDY	Maurer	DayMet
Mean filling	2.798/0.616	0.765/0.560	0.689/0.368
Last filling	2.803/0.649	0.705/0.422	0.693/0.253
M-RNN	—/—	0.616/0.413	0.798/0.494
BRITS	—/—	0.490/0.288	0.586/0.221
Transformer	0.311/0.215	0.354/0.194	0.369/0.113
TSEst	0.297/0.185	0.303/0.156	0.302/0.093

For all three datasets, TSEst outperforms the baselines in most cases. We could not run M-RNN and BRTIS on our available resources with TEDDY data because of its high dimensionality and subsequent huge memory requirement. These RNN based models work in the original high dimensional feature space whereas attention-based models used lower dimensional embeddings to estimate the missing values. This is a crucial drawback for the RNN-based models as many omics data like gene expression which is an essential part of enhancing modern medicine are high dimensional. Moreover, for Maurer and DayMet datasets, RNN-based models fall behind in performance compared to attention-based models. This is partly caused by the long-time dependencies of 500

time steps in these datasets that RNN struggles to accurately handle [186].

Table 5.5: RMSE/MAE scores for partially missing samples at different missing rates on Maurer data [random missing]

Method	20%	30%	40%	50%	60%	70%	80%	90%
Transformer	0.232/0.089	0.255/0.098	0.258/0.108	0.271/0.121	0.310/0.141	0.332/0.160	0.360/0.186	0.444/0.228
TSEst	0.238/0.088	0.241/0.096	0.242/0.104	0.265/0.118	0.299/0.131	0.318/0.147	0.330/0.163	0.401/0.205

In Table 5.5, we report the imputation performances for Maurer data at different missing rates and investigate how the missing rate impacts the uni-modal transformer and our multi-modal framework differently. We choose transformer as the baseline for this experiment based on the results in Tables 5.3 and 5.4 which show that only the transformer performs close to our proposed model. For missing rates between 20%-50%, two models perform similarly with our model slightly edging the baseline. It should be noted that our model and transformer use the same self-attention architecture, therefore would perform comparably given the same input. For low missing rates, available training data in the time series is enough to train the model and benefited moderately from the inclusion of another data modality. However, at missing rates higher than 50%, we can see a difference in imputation accuracy that can be attributed to the additional information from the cross-sectional data. To illustrate the contribution of the cross-sectional data in more detail, we extract the weights α and β calculated in the weighted addition block for test set. The weights represent how much importance is put on cross-sectional data vs the time series data for generating the imputed values. Figure 5.2 shows the distribution of the weights using kernel density estimate (KDE) plot at missing rate=50%, 70%, and 90%. The plot for cross-sectional data at 70% and 90% missing rate shift right signifying larger weights put on the cross-sectional data during imputation. Therefore, adding a cross-sectional data can improve imputation at higher missing rates which is often found in many real world datasets. TSEst provides a generalized framework for time series imputation that can work in both multi-modal and uni-modal settings. Moreover, if

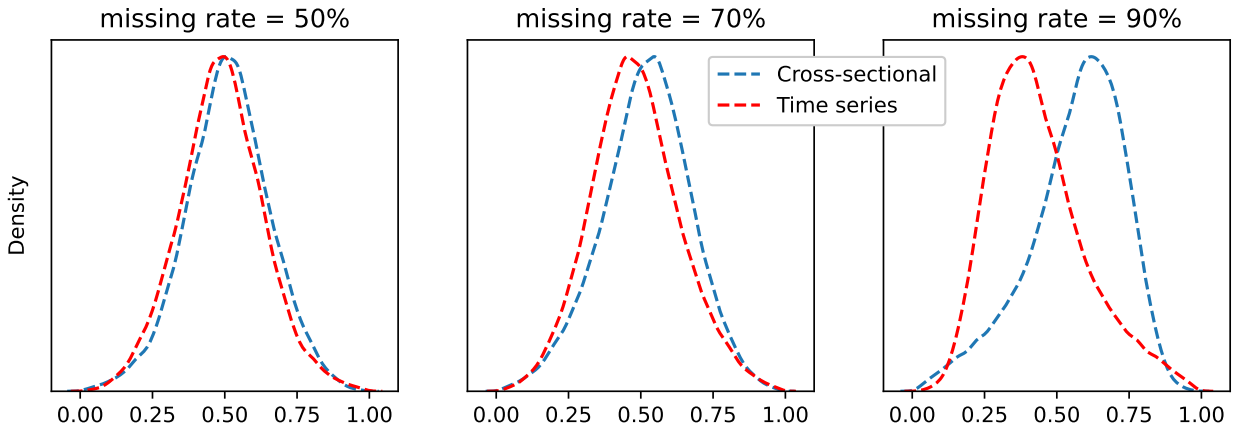

Figure 5.2: Weight distribution at different missing rates

additional modality has a low correlation with the incomplete time series in a multi-modal imputation, TSEst can automatically filter out unnecessary information from additional modality using weighted addition.

5.4.4 Imputation of completely missing samples

We extend the framework to impute the time series data for samples with no prior data available to investigate whether cross-sectional data can generate a reliable estimation of missing values by itself. For this experiment, training samples have both cross-sectional and time series data whereas validation and test samples only contain the cross-sectional data. The model is trained following the same procedure as the partially missing samples, except a larger value of γ and μ in equation 5.8. Artificial random masking is used to remove 40% of training data for a more robust learning and 100% of available data in validation and test samples for evaluation. During testing, we only enable Component 1 and set $\boldsymbol{Z}_3 = \boldsymbol{Z}_1$. RMSE and MAE scores of this experiment for TEDDY data are 0.320 and 0.213 respectively. The scores are comparable to imputation accuracy at 20% missing rate as shown in Table 5.3. SNPs present in the sites for DNA methylation, transcription

Table 5.6: RMSE/MAE scores for the completely missing samples at different length of Maurer data

Method	20	50	100	200	300	400	500
TSEst	0.555/0.331	0.572/0.351	0.625/0.389	0.598/0.377	0.611/0.382	0.583/0.365	0.564/0.355

factor binding, or miRNA targets can alter the gene expression level. For this reason, SNP has been successfully used before to predict cross-sectional gene expression [187]. Our results indicate that the same performance can be extended to impute time series gene expression as well using the proposed framework TSEst.

Another set of results for Maurer data is reported in Table 5.6 with different lengths of the time series ranging from 20-500 time steps to reflect how length impacts the ability of the cross-sectional data to generate missing values. Unlike TEDDY, scores drop for Maurer data which can be attributed to the stronger relation between SNP and gene expression that is absent in the cross-sectional and time series data in CAMELS. However, the imputation accuracy is better than mean filling, last filling, and M-RNN. It can be inferred from the results that cross-sectional data can generate reasonable time series with arbitrary length without a significant drop in performance with length increase.

5.4.5 Imputation using cross sectional vs time series data

We have shown the time series imputations using cross-sectional data in the above experiments. However, multiple time series data can also be available for the same cohort such as Maurer and DayMet in CAMELS dataset. In this experiment, we investigate how the model performs if we use a time series instead of cross-sectional data to impute another time series. We impute the Maurer data with 20% missing values using fully available DayMet data and present the results in Table

5.7. An LSTM architecture is used in place of the fully connected feed forward neural network (FFNN) in Component 1 to better characterize the time dependency. As the results show, using time series data does not provide any additional benefit.

Table 5.7: RMSE/MAE scores for time series-time series imputation

Imputation method	random	chunk
Cross-sectional–time series	0.238/0.088	0.303/0.156
Time series–time series	0.232/0.089	0.316/0.158

5.4.6 Model analysis

We propose an adaptive and interpretable weighted addition (WA) technique for merging the two time series data. We compare our proposed method with two baselines. The first one is directly adding the two time series data into a single dataset. The other one is concatenating the two datasets and then linearly transforming them to match the dimension of the original time series data. The results at different missing rates are tabulated in Table 5.8. Proposed weighted addition shows better performance in most cases. At a lower missing rate, all methods perform competitively,

Table 5.8: RMSE/MAE scores for model analysis

Missing rate	WA	Addition	Concatenation
20%	0.238/0.088	0.243/0.092	0.234/0.086
50%	0.265/0.118	0.299/0.135	0.297/0.129
90%	0.401/0.205	0.447/0.237	0.432/0.232
100%	0.564/0.355	0.618/0.406	0.630/0.405

however at a higher missing rate, there is a significant difference in imputation accuracy.

5.5 Summary

We proposed TSEst, a self-attention-based time series missing value imputation framework. TSEst uses a multi-modal architecture, exploiting an additional data modality for the imputation. An adaptive weighted addition technique assigns appropriate weights to each data modality for best imputation accuracy. Using two datasets, we showed that TSEst can effectively impute time series data using another cross-sectional data collected for the same set of samples and achieve better performance compared to state-of-the-art models from the literature. Moreover, it can impute time series values for samples with no prior available time series data. We demonstrated that the weighted addition mechanism provides interpretable insight into the time dependency of the datasets.

CHAPTER 6: USE OF MULTI-MODAL MISSING VALUE IMPUTATION IN TYPE 1 DIABETES STUDY

The work in this chapter has been published in the following paper:

Khandakar Tanvir Ahmed, Sze Cheng, Qian Li, Jeongsik Yong, and Wei Zhang. (2023). "Incomplete time-series gene expression in integrative study for islet autoimmunity prediction." Briefings in Bioinformatics, 24(1), bbac537. [188]

In previous chapter, we have shown the possibility and theoretical accuracy of multi-modal time series data imputation. The results were measured in terms of mean squared error/ mean absolute error loss without concrete proof of how they would behave in real prediction scenarios. In this chapter, we take this idea of multi-modal time series imputation to predict disease prognosis and aim to show that the imputed values are meaningful replacements for the missing values. Our imputed data not only allows us to predict prognosis for all patients regardless of their prior time series data collection status, it also improves the state-of-the-art prediction results in literature.

6.1 Introduction

Gene expression changes throughout the timeline of chronic diseases such as diabetes, hypertension, obesity, and heart disease; therefore, a periodically measured gene expression may better explain the underlying mechanisms of these diseases compared to cross-sectional gene expression collected once per participant [189]. Some prospective longitudinal cohort studies collect that information. However, these studies tend to suffer from loss to follow up [190, 191], which means the time series data will have missing values if participants are absent during scheduled

visits when data is collected. Moreover, limited by cost and logistics, data is often collected for a subset of participants, i.e., some participants will have no gene expression data available. An effective data imputation technique is necessary to use the gene expression for downstream analyses [33, 192]. Researchers have investigated computational methods for handling the missing value problem in gene expression, and several algorithms have been proposed to impute gene expression. The missing gene expression problem can be broadly divided into two groups: 1) contains participants with partially available gene expression, and 2) contains participants with no available gene expressions. Many frameworks have been developed to solve the prior stated problem that consider global or local relations among genes, domain knowledge, and other omics data for imputation [193, 68, 194, 82, 195]. The second group of missing value problems is more apparent in multi-omics analysis, where some participants can be present in another omics type but absent in gene expression. For such conditions, several frameworks have been developed that use other omics data to guide the imputation of gene expression [39, 196, 197]. As most studies evaluate gene expression profiles at a single time point, most of the available imputation frameworks are also designed to impute such gene expression datasets. The imputation of time series data offers additional challenges because of the time dependency among the time steps from the same participants. A handful of frameworks were proposed for the imputation of time series gene expression data [198, 195, 199] but they do not involve multi-omics data and participants with completely missing gene expression. More recently, some advanced algorithms have been proposed for time series data imputation in other domains [173, 200, 185, 32]. However, samples with no available gene expression still complicate integrative time series analysis to study chronic diseases.

Type 1 diabetes (T1D) is a common chronic disease in children caused by the autoimmune response against pancreatic β cells. Despite active research, the exact causes or any cure for the disease is still unknown [201]. Islet autoimmunity (IA), which precedes the clinical onset of T1D [202] can be used as a marker to study the progression towards T1D. The Environmental Determinants

of Diabetes in the Young (TEDDY) is a longitudinal prospective study that uses a nested case-control cohort to identify risk factors associated with T1D. Early and accurate identification of high-risk children (children with a high probability of developing IA) will allow us to design a better case-control cohort to identify risk factors, which may eventually lead to the prevention and cure of the disease. Therefore, predicting IA has been at the center of attention in diabetes studies for a long time [203, 169, 204]. Recent attempts to predict outcomes in T1D studies have used genetic factors [205, 206, 207, 208, 209], metabolic status [210, 211], family history, and environmental risk factors [203, 212, 169] for the prediction. Gene expression of the participants has been widely ignored, even though the predictive power of gene expression is well established in the literature to study different diseases [213, 214, 215, 37]. Integration of gene expression with other omics profiles is also well documented to result in improved prediction results for different objectives such as biomarker identification, patient stratification, and survival prediction [84, 216, 217, 218, 219] which may translate into a better outcome prediction for T1D. One reason for the reluctance to use gene expression is its weak association with the outcome, partially contributed by the high missing rate. Few previous studies [168, 169] have used time series gene expression from TEDDY to explain the progression towards T1D and found encouraging results for the T1D onset prediction. However, they only predict T1D for a small subset of total participants, and some use IA information for the prediction. IA information becomes available years after the birth of the child and their enrollment in the study. Therefore, the prediction can only be performed after a certain time. Missing data in gene expression hinders the analyses in these studies as well. TEDDY collects times series gene expression from its participants as well as their SNP, HLA genotype, and family history. The gene expression is collected for less than 6% of the enrolled participants and suffers from a large amount of missing time steps. It limits the opportunity for a comprehensive integrative study involving all participants. However, SNP, a cross-sectional data, is available for all participants, enabling us to impute partially or completely missing gene expression profiles.

6.1.1 Contribution

The primary objective of this work is to propose a model that will impute partially or entirely missing gene expressions with synthetic data. We employ a deep learning-based model to generate synthetic gene expression from SNP data and available gene expression. We demonstrate that it contains a competitive predictive signal compared to the true gene expression and improves state-of-the-art prediction results. We also explore the importance of time series gene expression in capturing the underlying mechanisms of T1D. The rest of the manuscript is organized as follows: TEDDY study setup, our research design, and methodology are described in the next section. The Experiments section is dedicated to experimental setups and validation of the results. The Discussion section contains a brief discussion of the results along with our limitations and future directions. Concluding remarks are presented in the last section.

6.2 Methods

6.2.1 Data sets and participants

TEDDY study is designed to identify the environmental risk factors impacting the development of IA and the onset of T1D [40]. TEDDY enrolls 8,676 high-risk children in this study based on the HLA genotype of the children and their first-degree relatives [203]. Follow-up for each child starts at three months and lasts until 15 years of age. Children are tested for islet autoantibodies (IAA, GADA, IA-2A, ZnT8A) at each visit, and gene expression is also measured. Visits for the participants are three months apart for the first four years. After that, it is three months for participants with any positive islet autoantibody test and six months for the rest. The outcome of interest in this study, IA, is defined as the presence of two consecutive positive tests for any particular islet autoantibody. In other words, if there are consecutive positive tests for at least

one of the four autoantibodies, we consider that participant as IA positive. Many risk factors, including HLA genotype, SNPs, dietary factors, family history, sex, and seroconversion age, have been investigated in previously published works that narrow down the candidates for a predictive study [220, 203, 221, 222, 168]. Although these risk factors are found to be weakly associated with T1D outcome [203], these studies ignored time series gene expression which may introduce

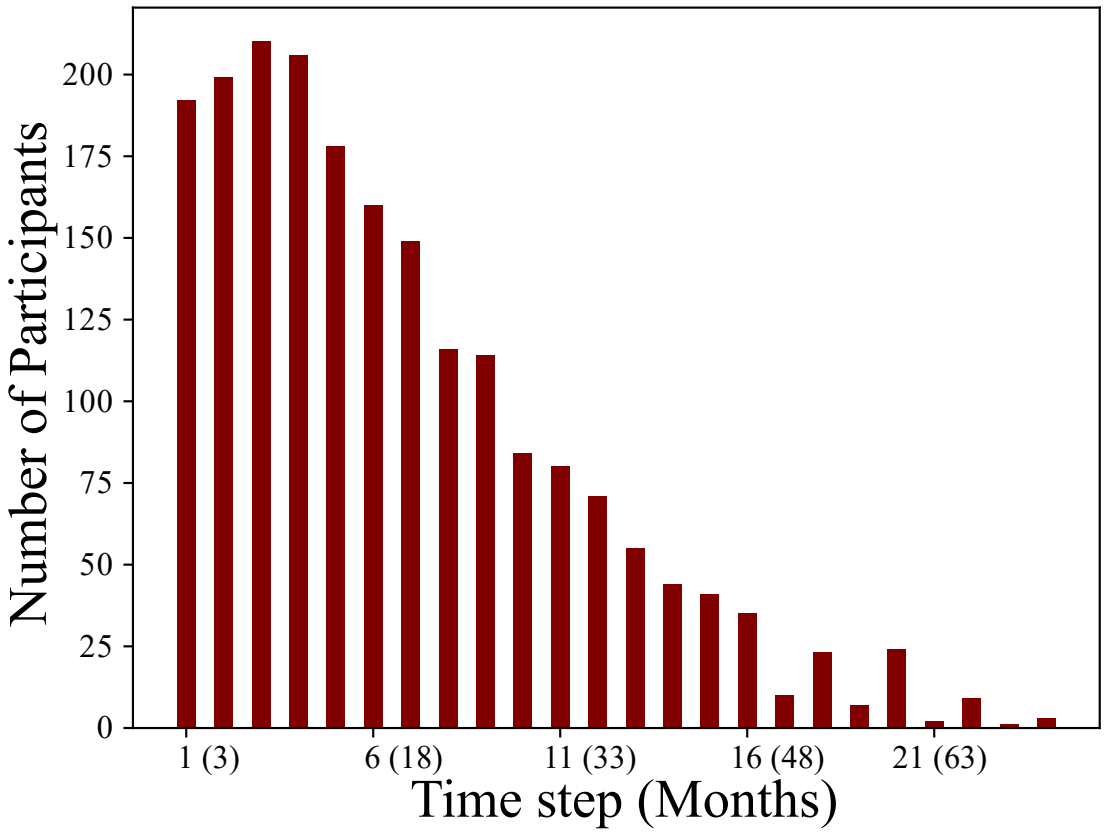

Figure 6.1: The number of available participants at each time step. The number of participants with available gene expression at each time step up to 24 time steps (72 months) are plotted. The plot shows a decrease in the availability of gene expression at later time points. 16^{th} time step is selected as an optimum point for gene expression cutoff.

complementary information and time factor for better prediction results.

Many TEDDY-identified risk factors have been previously explored, and family history, HLA genotype, and SNP were shown to be better predictors for IA status [203]. Based on the liter-

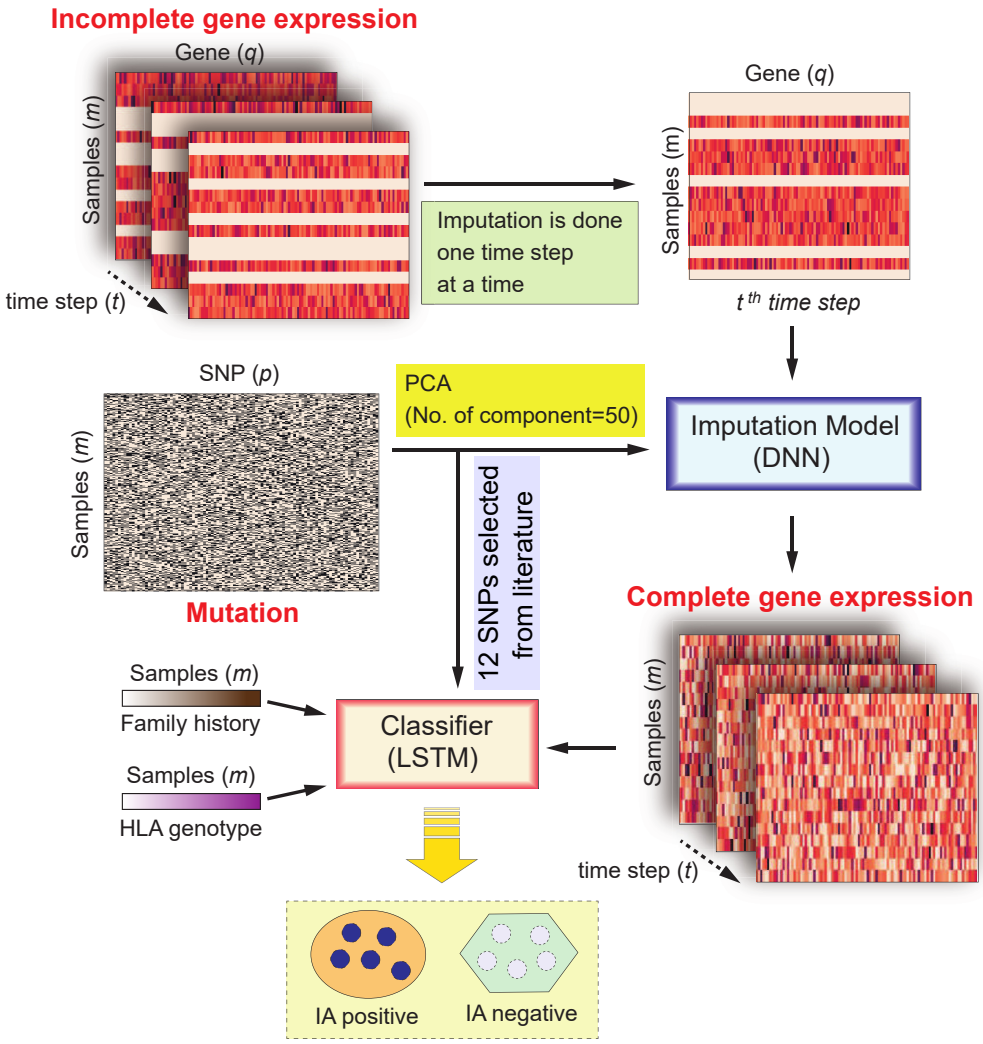

Figure 6.2: An overall illustration of the proposed framework. Incomplete gene expression is imputed using SNP in the imputation model (DNN). Completed gene expression, SNP, HLA genotype, and family history are fed into the classifier (LSTM) to predict IA positive and IA negative participants.

ature, we include 12 SNPs, HLA genotype, and family history in this study. Details about the 12 SNPs can be found in [203]. We performed an exhaustive search for the best SNP combination and found *rs4597342*, *rs12708716*, *rs4948088*, and *rs1143678* combined with HLA genotype and family history to be the best performing combination for IA status prediction. Therefore, we include these variables in further analyses of this study. Risk factors are binarized before feeding them into models. Family history was categorized as first-degree relatives having T1D vs. no T1D. SNPs were categorized as major (no copy of minor allele) vs. minor (one or two copies of minor alleles). The HLA genotype is defined as DR3/DR4 vs. others. TEDDY participants (m = 6,812) with available family history, HLA genotype, and SNP data are included in this study.

The gene expression in TEDDY is a time series with 2,013 time steps belonging to 401 children. Gene expression is collected until 72 months at 3 or 6 months intervals. Approximately 79% of time steps are missing for the 401 participants, which significantly impedes its ability to be used in a time series study. In the cohort of 6,812 participants, the missing rate rises to 98.77%, as the other 6,411 (94.11% of 6,812) participants have no available gene expression. Therefore, the gene expression is unusable for downstream analyses involving a cohort of 6,812 participants. Nevertheless, the number of missing participants changes across time steps, allowing us to remove some time steps with fewer available participants.

The number of available participants at each time step is presented in Fig. 6.1 which illustrates that the rate of missing participants increases in later time steps. Moreover, after 48 months, some participants visited every six months instead of 3, resulting in an even lower data availability rate. As available data is necessary to train the imputation model, a lower data availability rate disrupts the model training; thus, the quality of the synthetic gene expression. To reduce the impact of missing data and maintain a regular interval of 3 months between consecutive time steps, we set a cutoff of 48 months for gene expression in this study. Therefore, gene expression of each participant consists of 16 time steps corresponding to 3 to 48 months at three months intervals.

Table 6.1: Dimensions of gene expression and SNP used in different stages of the study

Dataset	Dimension
Original gene expression	$401 \times 17,039$
Imputed gene expression	$6812 \times 17,039$
Original SNP	$6812 \times 176,586$
SNP (for imputation)	Top 50 PCs of original SNP
SNP (for IA prediction)	12 SNPs selected from literature

Although setting a cutoff lowers the missing rate to 98.22%, it is still impractical to use gene expression with predictive algorithms without an effective data imputation. Therefore, we propose a deep learning-based imputation model described in the following subsection that can generate synthetic gene expression at missing time steps from SNPs. We keep 17,039 protein-coding genes in the gene expression. 17,039 features may overfit the model or impose a computational burden with redundant information [223, 224]; so we find an optimal number of genes using forward feature selection that will provide us with the best prediction results. Once we have the optimal number of genes, gene expression, family history, and SNPs are merged into a single time series dataset. For family history, HLA genotype, and SNPs, the same value for a participant is replicated at every time step. Dimensions of the datasets used in different stages of this study are tabulated in Table 6.1.

6.2.2 Imputation model overview

The overall framework of the proposed study is illustrated in Fig. 6.2. The framework has two main components: a deep learning-based imputation model and a long short-term memory (LSTM) based classifier. Synthetic gene expression is first generated for missing time steps through the imputation model using SNP and available gene expression. Family history, HLA genotype, SNP,

and completed gene expression are then fed into the classifier to predict IA positive and IA negative participants.

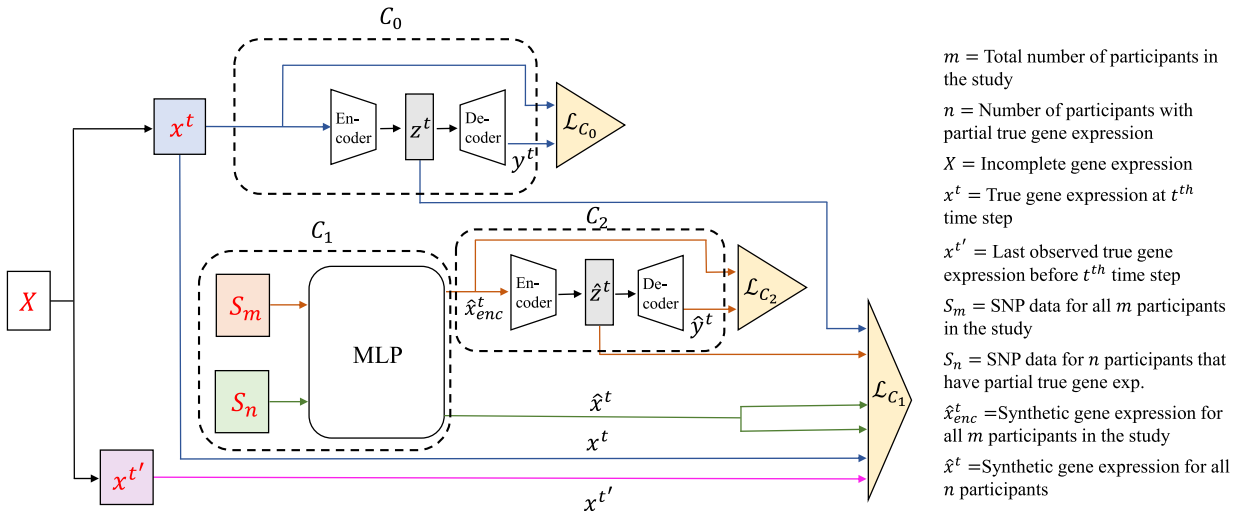

Figure 6.3: An illustration of the proposed imputation model. Incomplete gene expression $\boldsymbol{X}$ is imputed using autoencoders C_0, C_2 and multilayer perceptron (MLP) C_1.

Although gene expression is either partially or completely missing for every participant, SNP data is available for all of them. Therefore, our proposed imputation model is trained to map the SNP data to gene expression and generate the value for missing time steps. The imputation is carried out for each time step separately, i.e., the model is retrained for imputing every time step as seen in Fig. 6.2. For imputing a time step, participants with available gene expression at that time step are separated and randomly divided into training and validation sets with a 70-30 split ratio. All other participants without gene expression are considered as the test set. The training samples' SNP data and gene expression are used as input and output to train the model. The imputation model is illustrated in Fig. 6.3. Let m be the total number of participants in the study. SNP data ($\boldsymbol{S}$) is available for all m participants, whereas the gene expression is available for n participants among them. q genes from n participants in the gene expression $\boldsymbol{X} \in \mathbb{R}^{n \times T \times q}$, observed in

$t = (1, 2, ..., T)$ is defined as $\boldsymbol{X} = [\boldsymbol{x}^1, \boldsymbol{x}^2, ...\boldsymbol{x}^T]$ where $\boldsymbol{x}^t \in \mathbb{R}^{n \times q}$ denotes the observation from t^{th} time step. For imputation of t^{th} time step, we first train an autoencoder (C_0) to find a lower dimensional representation of $(\boldsymbol{x}^t)' \in \mathbb{R}^{q \times n}$, given by $\boldsymbol{z}^t \in \mathbb{R}^{q \times h}$ where $()'$ represents transposition and h is the embedding size. $\boldsymbol{z}^t$ contains the property of each feature in the observed data at t^{th} time step, which will be later used in equation 6.7 to guide the synthetic data generation. Both encoder and decoder are five layers feed-forward neural networks. The encoder finds the lower dimensional embedding from the original data, which is fed to the decoder as it tries to reconstruct the original data from the embedding. The autoencoder is trained with the reconstruction loss for 100 epochs using Adam optimizer and a learning rate of 0.0001. Output from the encoder and decoder are given by equations 6.1 and 6.2 respectively and the network is trained following equation 6.3.

$$\boldsymbol{z}^t = Encoder(\boldsymbol{x}^t) \tag{6.1}$$

$$\boldsymbol{y}^t = Decoder(\boldsymbol{z}^t) \tag{6.2}$$

$$\mathcal{L}_{C_0} = \|\boldsymbol{y}^t - \boldsymbol{x}^t\|_2^2 \tag{6.3}$$

Then we move forward to the imputation of missing values. It consists of two components: i) a six-layer fully connected deep neural network (C_1) and ii) an autoencoder (C_2) that follows the first component (C_1). C_1 takes SNP data $\boldsymbol{S}_n \in \mathbb{R}^{n \times p}$ as input and generates synthetic gene expression $\hat{\boldsymbol{x}}^t \in \mathbb{R}^{n \times q}$ for t^{th} time step. p is the number of features in the SNP data. Dimension of the SNP data is reduced to avoid overfitting using principal component analysis (PCA) implemented using sklearn.decomposition.PCA package [225]. The top 50 principal components (PCs) are used in the imputation. $\hat{}$ notation is used throughout the study to denote synthetic data and values derived from synthetic data. The ReLU activation function follows hidden layers in the C_1 network. Layers can be formulated as:

$$\boldsymbol{x}_{out}^t = \delta(\boldsymbol{W}\boldsymbol{x}_{in}^t + \boldsymbol{b}) \tag{6.4}$$

where $\boldsymbol{W}$, $\boldsymbol{b}$ are learnable parameters and δ is the activation function. For the first layer, $\boldsymbol{x}_{in}^{t} = \boldsymbol{S}_n$ and in the final layer $\boldsymbol{x}_{out}^{t} = \hat{\boldsymbol{x}}^{t}$. Additionally, we use the same model to generate the synthetic gene expression $\hat{\boldsymbol{x}}_{enc}^{t} \in \mathbb{R}^{m \times q}$ for all m participants from $\boldsymbol{S}_m \in \mathbb{R}^{m \times p}$. To recollect, m is the total number of participants in the study and n is the participants with available gene expression; therefore, $\boldsymbol{x}_{out}^{t} \subset \hat{\boldsymbol{x}}_{enc}^{t}$. Afterwards, $\hat{\boldsymbol{x}}_{enc}^{t}$ is fed into the autoencoder C_2 where an embedding $\hat{\boldsymbol{z}}^{t}$ is generated that represents the characteristics of the imputed features in a lower dimension following equations 6.5 and 6.6. The purpose of this autoencoder C_2 is to ensure that feature properties remain the same before and after imputation, which means generated data will have similar properties as the true data.

$$\hat{\boldsymbol{z}}^{t} = Encoder(\hat{\boldsymbol{x}}_{enc}^{t}) \tag{6.5}$$

$$\hat{\boldsymbol{y}}^{t} = Decoder(\hat{\boldsymbol{z}}^{t}) \tag{6.6}$$

The objective functions for C_1 and C_2 are formulated as equations 6.7 and 6.8 respectively.

$$\mathcal{L}_{C_1} = \|\hat{\boldsymbol{x}}^{t} - \boldsymbol{x}^{t}\|_2^2 + \frac{1}{e^{d}}\|\hat{\boldsymbol{x}}^{t} - \boldsymbol{x}^{t'}\|_2^2 + \|\hat{\boldsymbol{z}}^{t} - \boldsymbol{z}^{t}\|_2^2 \tag{6.7}$$

$$\mathcal{L}_{C_2} = \|\hat{\boldsymbol{y}}^{t} - \hat{\boldsymbol{x}}_{enc}^{t}\|_2^2 \tag{6.8}$$

The first element ($\|\hat{\boldsymbol{x}}^{t} - \boldsymbol{x}^{t}\|_2^2$) in equation 6.7 makes the synthetic gene expression for the n participants similar to the true gene expression at t^{th} time step. The second element ($\frac{1}{e^{d}}\|\hat{\boldsymbol{x}}^{t} - \boldsymbol{x}^{t'}\|_2^2$) introduces information from previous time steps in the imputation. $\boldsymbol{x}^{t'}$ represents the last observed gene expression at $t^{'}$ time step while imputing data at t^{th} time step. We assume that gene expression at a time step is more similar to its closest time step, which provides a better estimation for the missing gene expression. d denotes the time difference between t^{th} and $t^{'}$ time steps which ensures that gene expression observed in closer time step from t^{th} has more contribution in the imputation

compared to gene expression observed at further time steps. The last element ($\|\hat{\boldsymbol{z}}^t - \boldsymbol{z}^t\|_2^2$) in equation 6.7 ensures that feature characteristics are similar in gene expression ($\boldsymbol{X}$) before and after imputation.

C_1 is trained for 100 epochs using Adam optimizer with a learning rate of 0.001, and C_2 is trained for 25 epochs using Adam optimizer with a learning rate of 0.00001. C_2 is fully trained at each epoch of C_1 to obtain the best embedding value. High-quality embedding generated by C_2 will in turn result in better training for C_1 as seen in equation 6.7.

As most participants have no true gene expression, the imputation model must have two characteristics to ensure the best performance. It has to be able to generate synthetic data for a participant using only SNP and maximize the information extraction from the SNP simultaneously. In our model, the multilayer perceptron, C_1, is responsible for mapping SNP to gene expression. To ensure that C_1 uses only SNP as input, we can not integrate available gene expression from other time steps. On the other hand, ignoring other time steps will result in loss of valuable information and inferior mapping of SNP to gene expression that contradicts the second characteristic. As mentioned before, all participants in the training set have partial true gene expression. Therefore, we employ the available gene expression from other time steps for a participant in the objective function through the second element, whereas once trained, C_1 only uses SNP data to generate synthetic gene expression. It ensures that we can generate synthetic data for participants in the test set with no prior gene expression and also harness the prior information from participants during training. The validation set is used to tune the hyperparameters and choose the best model during training. Then SNP data of the test set is fed into the network to generate gene expression values for those participants with missing time steps. The same procedure is repeated for each time point. The deep neural network model is implemented in PyTorch [226].

6.2.3 Classifier and metrics

In this study, we use a long short-term memory (LSTM) based classifier [227] for time series predictions. LSTM is a type of recurrent neural network that takes time series data as input and maps that to a label considering the time factor in the analysis. It is a three-layer network followed by a fully connected layer and a sigmoid activation function. The hidden size of the LSTM is 200. For predictions using only family history, HLA genotype, and SNP, we use random forest implemented through sklearn.ensemble.RandomForestClassifier package. Area Under the Receiver Operating Characteristic Curve (AUC), sensitivity, specificity, and Youden's index are applied to evaluate the performance of the classifiers.

6.3 Results

A total of 6,812 samples from TEDDY with family history, HLA genotype, and SNP data available are included in our Study. 338 (4.96 %) of them develop IA within 24 months. In the experiments, we show the proposed improvement of IA prediction after the integration of gene expression with family history, HLA genotype, and SNP, the quality of our imputed gene expression used in the prediction, and the enriched gene sets (e.g., pathways) are significant for T1D pathogenesis.

6.3.1 Integration of gene expression improves IA prediction

6.3.1.1 Feature properties and selection

6.3.1.1.1 Family history, HLA genotype, and SNPs

TEDDY identified risk factors have a wide range of abilities to predict IA. Sensitivity, specificity, Youden's index, AUC of family history, HLA genotype, and SNPs used in this study for predicting IA outcome at 24 months are reported in supplementary Table S1. Family history is the best predictor of IA (Youden's index = 0.265) followed by HLA genotype, *rs4597342*, *rs12708716*, *rs4948088*, and *rs1143678* respectively.

6.3.1.1.2 Gene expression

An optimum number of genes from the gene expression is added to the family history, HLA genotype, and SNP data to improve the IA prediction. The genes are ranked based on their variance across the samples, and genes with the highest variances are added sequentially to the LSTM-based classifier along with family history, HLA genotype, and SNP until its performance on validation data goes down. Based on this result, we choose to add the top 10 genes in our analysis which gives the highest validation AUC of 0.73.

6.3.1.2 Prediction results

6.3.1.2.1 Improved IA outcome prediction

A collective role of omics layers determines the physiology behind a complex disease like T1D. Therefore, integrating additional omics data into the analysis can provide complementary information enabling a better outcome prediction. We predict the IA status of participants using: 1) family history, HLA genotype, and SNPs, 2) synthetic gene expression, 3) combination of family history, HLA genotype, SNPs, and synthetic gene expression. IA status labels are generated at different time cutoff $t = [24, 30, 36, 48, 72]$ months, individually, where all participants developing IA by t^{th} month are considered as IA positive, and all others are considered IA negative. Predictions using only gene expression and a combination of family history, HLA genotype, SNPs, and gene expression are carried out employing the LSTM model described in the subsection 6.2.3. On the other hand, predictions using family history, HLA genotype, and SNPs are carried out employing the

Table 6.2: Predictions at different IA cutoff. Results (sensitivity, specificity, Youden's index, AUC) of IA status prediction using three input data at different IA cutoffs are calculated. The combination of family history, HLA genotype, SNP, and gene expression shows better performance compared to them individually. AUC, sensitivity, and Youden's index drop when the IA cutoff is increased suggesting the difficulty associated with predicting further into the future. Improvements using combined data at all cutoffs are statistically significant (p-value$<$0.001).

t	Family history+HLA+ SNP				Gene Expression				Combined			
	Sen	Spe	Y index	AUC	Sen	Spe	Y index	AUC	Sen	Spe	Y index	AUC
18	0.597	0.701	0.298	0.651	0.421	0.799	0.220	0.623	0.640	0.750	0.390	0.717
24	0.542	0.719	0.261	0.643	0.393	0.861	0.254	0.639	0.622	0.761	0.383	0.715
30	0.484	0.746	0.230	0.634	0.390	0.871	0.261	0.639	0.599	0.771	0.370	0.708
36	0.467	0.737	0.204	0.623	0.378	0.887	0.265	0.646	0.575	0.772	0.347	0.701
48	0.476	0.716	0.192	0.598	0.342	0.910	0.252	0.633	0.531	0.780	0.311	0.681
72	0.460	0.715	0.175	0.591	0.360	0.876	0.236	0.631	0.494	0.784	0.278	0.671

random forest model described in the subsection 6.2.3. All predictions in this study are repeated 50 times with random splitting of samples into training, validation, and test set. The mean values of AUC, sensitivity, specificity, and Youden's index of test sets from 50 repetitions are reported in Table 6.2 and all results thereafter.

The results in Table 6.2 illustrate the improvement in prediction at every time cutoff caused by the inclusion of gene expression with other features, which signifies the importance of additional information contained within the gene expression. Moreover, gene expression considers the time factor and reflects the physiological changes over a period of time instead of a snapshot of the underlying processes. We also find it more difficult to predict further into the future as sensitivity, Youden's index, and AUC decreases gradually with a higher cutoff value of t. AUC, sensitivity, specificity, and Youden's index of our proposed model show better results than the baseline where we used only the time-invariant features. Higher sensitivity is crucial for this prediction as false negative results can result in neglected care of a high-risk child. Moreover, for IA status cutoff at 24 months, our proposed model (AUC 0.715) outperforms the state-of-the-art result published by the TEDDY study group in an 8-year progress report [203] (AUC 0.682). They also used family history, HLA genotype, and SNPs in the predictive model; therefore, it can be inferred that the improvement in our study is caused by the use of reliable synthetic gene expression. Additionally,

Table 6.3: Predictions of different IA outcomes. Results (sensitivity, specificity, Youden's index, AUC) of first islet autoantibody appearance at 24 months are calculated. The combination of family history, HLA genotype, SNP, and gene expression shows better performance compared to them individually.

t	Family history+HLA+ SNP				Gene Expression				Combined			
	Sen	Spe	Y index	AUC	Sen	Spe	Y index	AUC	Sen	Spe	Y index	AUC
IAA-first	0.486	0.721	0.207	0.635	0.454	0.798	0.252	0.627	0.615	0.716	0.331	0.690
GADA-first	0.524	0.738	0.262	0.657	0.351	0.905	0.256	0.622	0.646	0.759	0.405	0.718

we investigated the prediction of the appearance of the first Islet autoantibody type by 24 months. The results are tabulated in Table 6.3 which shows that combined data performs better at predicting both IAA-first (IAA appears first) and GADA-first (GADA appears first) participants compared to the baselines.

6.3.1.2.2 Impact of time series gene expression

As most studies collect single gene expression data from a participant, we designed an experiment to investigate what the results would be if TEDDY collected a cross-sectional gene expression in-

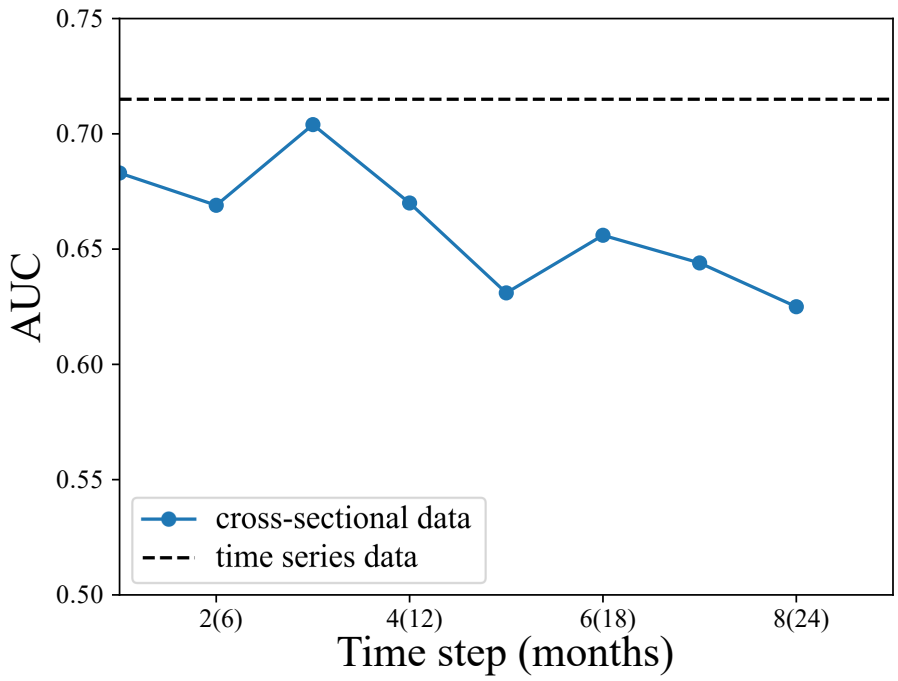

Figure 6.4: IA status prediction using one gene expression time step, family history, HLA genotype, and SNP. IA status is predicted at 24 months to illustrate the predictive ability of the gene expression if collected at one time point instead of a longitudinal study.

stead. We use the random forest to predict IA labels at 24 months using gene expression at one time step up to 24 months along with family history, HLA genotype, and SNP. The experiment is repeated for each time step; therefore, eight predictions correspond to the eight time steps (24 months) in gene expression. All predictions use the same value for family history, HLA genotype, and SNP, differing only in gene expression data. The results are illustrated in Fig. 6.4 which shows decreased performance at all time steps. The best AUC (0.704) is obtained when gene expression at 3^{rd} time step (9 months) is used to predict the IA status. The results show a significant gain in prediction performance by including the time factor of gene expression in the analysis. Moreover, the AUC values drop at later time steps, which indicates a deteriorating predictive signal at synthetic gene expression at later time steps. This behavior can be attributed to the insufficient availability of true gene expression (training data) in later time steps, as shown in Fig. 6.1, which results in a decaying training of the imputation model. However, our proposed model is not significantly vulnerable to this limitation as LSTM considers all time steps during the prediction.

Table 6.4: Predictions with gene expression at different time cutoffs. Results (sensitivity, specificity, Youden's index, AUC) of IA status prediction using gene expression and combined data up to t^{th} month are calculated. Higher value of AUC, sensitivity, and Youden's index when the cutoff is increased shows the improvement associated with additional time steps. * denotes the results with statistically significant differences compared to the result using all time steps (48^{th} months).

t	Gene Expression				Combined			
	Sen	Spe	Y index	AUC	Sen	Spe	Y index	AUC
12	0.317	0.945	0.262	0.608*	0.578	0.792	0.370	0.701*
24	0.357	0.907	0.264	0.623	0.577	0.800	0.377	0.710
36	0.357	0.904	0.261	0.631	0.502	0.787	0.289	0.721
48	0.421	0.799	0.220	0.623	0.622	0.761	0.383	0.715

6.3.1.2.3 *Impact of the availability of gene expression*

We used 16 time points in our time series data analysis which is determined by the regular interval and availability of gene expression up to 48 months. We investigate the importance of using longer time series data for prediction by setting different cutoffs to gene expression. For a cutoff at t^{th} month, we only use the input data up to t^{th} month in our LSTM-based classifier, which imitates the limitation in data availability. IA status for this experiment is generated at 24 months and is fixed for all gene expression cutoffs. The mean AUCs, sensitivity, specificity, and Youden's index, are reported in Table 6.4. Using combined input data with more time steps improves the prediction, which is expected as longer time series data can capture more physiological changes. Moreover, our proposed model shows a robust performance with limited data availability. The AUC drops to 0.701 from 0.715 (1.96% drop) even if we use 12 months as the cutoff for gene expression (25% of input data). We also investigate whether SNP solely contributes to the predictive ability of synthetic gene expression. In that case, we could use SNPs to replace gene expression and still get similar predictive performance. We find that SNPs by themselves have a poor prediction but can help us get an effective mapping to gene expression.

6.3.2 *Quality of synthetic gene expression*

The improvement in prediction with the addition of synthetic gene expression with family history, HLA genotype, and SNP depends on the quality of the synthetic data. Here we design an experiment to compare the predictive ability of synthetic gene expression against true gene expression. Only the 401 samples with true gene expression are included in this experiment to make the results comparable. Therefore, we have two sets of data; a true gene expression dataset and a synthetic gene expression dataset representing the same samples. We predict IA labels at different time cutoffs of $t = [18, 24, 30, 36, 42, 48]$ months using the two datasets using the LSTM based classifier.

The results are shown in Fig. 6.5 which illustrates the better predictive performance of synthetic data across all time step cutoffs. The improvement can primarily be attributed to the higher availability of data in synthetic gene expression. As mentioned before, 79% of time steps are missing in the cohort of 401 participants, which translates to 79% time steps having a synthetic gene expression against the 21% time steps having true gene expression. More time steps in the input time series data resulted in better analysis and, consequently, a higher AUC. However, true gene expression only comprises approximately 1.5% of the input time series gene expression for 6,812 participants, as most participants have no available true gene expression. It is inconsequential to merge the true gene expression with synthetic gene expression; therefore, all predictions in the subsection 6.3.1.2 were designed using only synthetic gene expression.

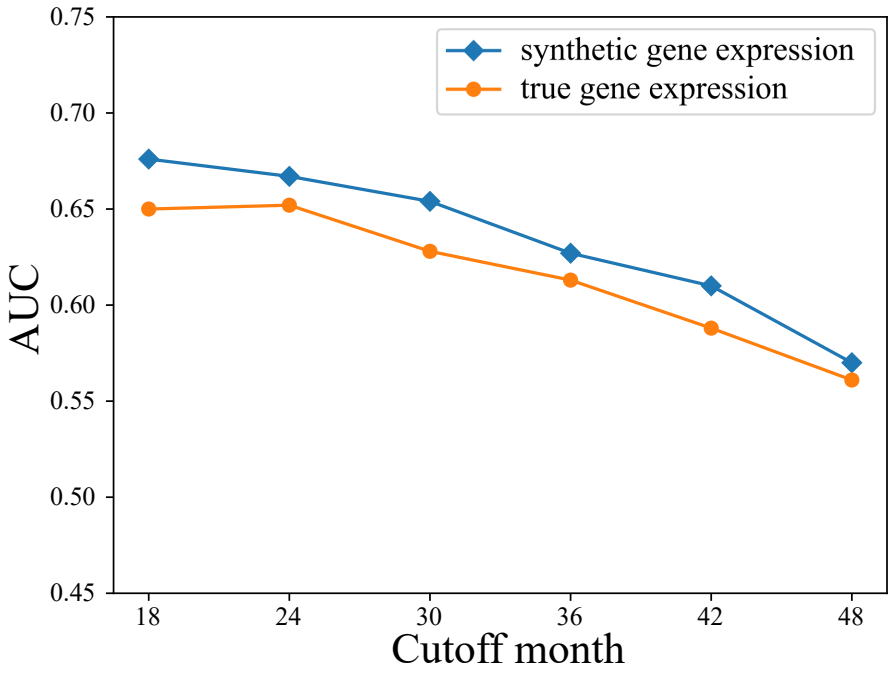

Figure 6.5: IA status prediction using true gene expression and synthetic gene expression. IA status is predicted using true and synthetic gene expression representing the same 401 participants.

6.4 Discussion

Detection of IA-positive children helps the researchers in the early identification and intervention of high-risk children. It also helps them to reduce the cohort size and increase T1D study efficiency. However, detection of IA might not always be enough as children can develop IA years after birth, whereas TEDDY participants were enrolled at birth. Hence, the prediction of IA can play a significant role in finding high-risk children more effectively. In T1D outcome prediction, the time series gene expression collected by TEDDY is inadequately used partly because of a high percentage of missing data. We generated synthetic gene expression from SNP to solve the missing data problem that shows competitive performance compared to true gene expression. We also successfully translated it into a better IA prediction, as shown in Table 6.2. Inspired by the superior predictive ability, we identified several pathways known to be related to T1D using synthetic gene expression reported in supplementary Table S5. Although gene expression with all true data might improve the prediction even further, due to the longitudinal data collection procedure limitations, missing data is also inevitable for future participants. This is evident because all 401 participants in the existing TEDDY study have incomplete gene expression. Therefore, developing a framework that can reduce the reliance on time series gene expression collection but accounts for the improvement introduced by the time factor is an important task. In this study, not only do we predict the IA status with higher sensitivty, specificity, Youden's index, and AUC, but we can do it with all synthetic gene expression in the classifier.

In our study, we used input data for up to 48 months but, in some cases, predicted IA status earlier than 48 months, as seen in Table 6.2. For example, two critical questions arise if we try to predict IA status at 24 months using input data up to 48 months. First, whether the prediction results are biased due to the presence of gene expression after 24 months. Secondly, participants are tested for islet autoantibodies when gene expression is collected at visits on or after 24 months. If the

participants already have their IA results, further prediction becomes a moot point. To address the first concern, in this study, all gene expressions used in classifications are synthetic data, and true gene expression is only used to generate them. We have shown better prediction results for 6,812 participants using true gene expression from only 401 participants. Moreover, the percentage of available gene expression is exceedingly tiny at 1.23%. Therefore, it can not create a significant bias for the prediction results. We also showed competitive results when IA status is predicted at 24 months using input data up to 12 or 24 months in Table 6.4. For the second concern, it is to be noted that we do not have any gene expression for approximately 94% of participants, which is the only time series data used as input in the classifier. For future participants, we do not need to collect gene expression; instead, the synthetic gene expression can be generated as soon as we have their SNP data and then predict their IA status years down the line. Incomplete RNA-seq gene expression is also available for the cohort; however, due to the inferior prediction performance we observed, microarray gene expression was used in this study.

Our proposed method improves IA prediction after including synthetic gene expression. We focused on gene expression in early life IA prediction and identifying prognostic genes, whereas family history, HLA genotype, and selected SNPs are used based on literature. A study involving other datasets such as metabolites, more SNPs, and environmental variables can further improve the prediction accuracy and draw a more detailed picture of the disease pathogenesis. Additionally, the participants in TEDDY are all high-risk children screened using the HLA genotype. Therefore, the true gene expression used as training data in the imputation model is also from those high-risk participants. Synthetic gene expression for children not yet identified as high risk can be inaccurate if the imputation model is trained using only the gene expression of high-risk participants. Therefore, our proposed pipeline is not an alternative to HLA genotype-based risk assessment but rather complements it to identify high-risk children better.

6.5 Summary

T1D is a chronic autoimmune disease characterized by irreversible destruction of islet β-cell. The incidence and prevalence of T1D have increased worldwide in recent years, which can disrupt the access and affordability of insulin, the only treatment to keep a T1D patient alive. Therefore, it is now more important than ever to find the key factors affecting the onset of this disease and develop effective treatments or cures. A comprehensive case-control study such as TEDDY can provide the researchers with answers to those questions. Early prediction of IA can help us design a better case-control study and ensure in-time care for high-risk children. This study offers an approach for generating synthetic time series gene expression from SNP and obtaining an improved and early IA prediction. Our proposed framework improves state-of-the-art IA prediction by integrating synthetic gene expression in the analysis. Additionally, we compared the time series gene expression against cross-sectional gene expression and showed superior performance of time series gene expression even when it is entirely synthetic. It also widens the door for further computational analyses to link genes to T1D outcomes and time series analyses using incomplete multi-omics data to study other chronic diseases. This chapter also provides tangible proofs of the effectiveness of multi-modal imputation frameworks. We not only predicted IA with better accuracy than previous studies, we can perform this prediction even when no gene expression was available.

CHAPTER 7: INTERACTION PREDICTION IN HETEROGENEOUS GRAPH

In the last chapter of this dissertation, we focus on interaction prediction in heterogeneous graph. The problem we have chosen is drug-target interaction prediction where the problem statement is analogous to inter-omics interaction prediction. An interaction prediction model can reduce the noise in the network and improve the performance of multi-omics integration.

7.1 Introduction

In the relentless pursuit of novel therapeutic agents, the intricate interplay between drugs and their biological targets has become the focal point of modern pharmaceutical research. The concept of drug-target interaction (DTI) constitutes the cornerstone of contemporary drug discovery and development, providing a fundamental framework for understanding the mechanistic foundations of pharmacological interventions. Amid the ever-evolving challenges posed by drug resistance and adverse drug reactions, the exploration of DTI not only expedites the identification of novel drug candidates but also augments our capacity to repurpose existing compounds for diverse therapeutic applications. Experimental assays have proven to be the gold standard for DTI identification [50]. However, research indicates that the expenses associated with the development of new drugs vary between $\$314$ million to $\$2.8$ billion, while the duration of clinical development typically spans between 8.2 to 10.0 years [228, 229]. These substantial investments in time and resources have made DTI prediction an indispensable tool to aid the initial stages of drug discovery by expediting the identification of potential drug-target interactions, thereby streamlining the process of lead compound selection and, consequently, experimental validation.

Numerous studies have demonstrated the utility of computational approaches, including machine learning algorithms, network-based methods, and molecular docking simulations for DTI prediction. In recent times, the advancement of DTI prediction has been notably accelerated, primarily attributed to the extensive accumulation and accessibility of biomedical datasets. This surge is further propelled by the remarkable progress of deep learning techniques, which have showcased exceptional success across diverse realms of scientific research and asserted themselves as the predominant method for DTI prediction. Several advanced deep learning-based frameworks for DTI prediction have emerged, utilizing diverse sets of data as input. These frameworks can be broadly categorized into knowledge graph-based methods [1, 41, 42], 3D structure-based approaches [45, 46, 47, 48], 2D pairwise distance map-based techniques [50, 51], and 1D sequence-based methods [52, 53, 54]. Heterogeneous knowledge graph (KG)-based methods have demonstrated success in various scenarios of DTI prediction, including warm start, cold start for drugs, and cold start for proteins. Cold start predictions involving unknown drugs or proteins are particularly challenging as limited or no information about that drug or protein is available during model training. Despite this challenge, KG-based models leverage semantic relationships with other entities (such as shared pathways, biological processes, or functional annotations) and diverse data sources, enabling them to achieve competitive performance in cold start predictions. However, it's crucial to note that KG-based methods demand large amounts of heterogeneous datasets and substantial computational resources to achieve state-of-the-art results. Their performance is also contingent on the completeness of the knowledge graph. Structure and sequence-based methods generally tend to perform worse for cold start predictions if the cold start protein or drug has no structural or sequential homologs with known interactions in training. Moreover, obtaining high-quality structural data for all proteins of interest can be challenging and time-consuming and requires significant computational resources. On the contrary, 1D sequences, such as amino acid sequences for proteins and SMILES (Simplified Molecular Input Line Entry System) for drugs, represent the most readily available form of input data and require less computation due to their

simplified representation. Ensuring the quality of data is also more straightforward compared to knowledge graphs and structural information. Therefore, addressing the limitations associated with cold start problems using 1D sequences holds the potential to accurately predict interactions for a broader spectrum of drugs and proteins compared to other methods.

The adoption of pretrained language models (LMs) has emerged as a transformative tool across a spectrum of research domains. BERT (Bidirectional Encoder Representations from Transformers) [230] brought about a paradigm shift in natural language processing tasks, and its impact extended to other domains such as ESM, ProtBert, and ProteinBERT [231, 232, 233] for protein feature extraction. Similarly, in drug-related contexts, models like ChemBERTa, ChemGPT, and MoL-Former [234, 235, 236] have played a crucial role in extracting drug features. These pretrained models have found applications and validation in previous DTI prediction studies, wherein embeddings are generated utilizing LMs [237, 238, 239]. These embeddings generated by LMs are independent, meaning no neighborhood information is considered during their generation. While such approaches have proven effective, recent studies, including those utilizing KG-based frameworks, have demonstrated the efficacy of neighborhood-based embedding generation for DTI prediction [240]. Incorporating neighborhood information into language model-based embeddings has the potential to yield improved representations for both drugs and proteins. Moreover, previous language model-based DTI prediction studies [237, 238] lack a comprehensive comparison with other methods, focusing only on the comparison among the language model variants.

In this chapter, we introduce a novel framework, DTI-LM, designed for predicting drug-target interactions by leveraging language models to generate encodings from protein amino acid and drug SMILES sequences. Going beyond traditional approaches, we enhance the encoding process by introducing graph attention networks (GAT). These networks enrich the representations of proteins and drugs with neighborhood information, thereby contributing to more nuanced and context-aware DTI predictions. Our experimental findings substantiate the effectiveness of the pro-

posed DTI-LM framework, demonstrating superior performance compared to existing state-of-the-art DTI prediction models while utilizing fewer data and computational resources. Furthermore, we investigate the current limitations associated with language model-based DTI prediction. This exploration allows us to gain insights into the challenges and boundaries that currently exist in protein and drug language models, providing a foundation for potential future enhancements and refinements in language model-based drug-target interaction prediction.

7.2 Methods

In this section, we first introduce the mathematical notations employed in this study, followed by the proposed framework, DTI-LM. The framework can take protein amino acid sequences and drug SMILES sequences as inputs in language models, followed by graph attention networks and a multi-layer perceptron (MLP) to predict DTIs. We then discuss the baselines used in this study to illustrate the improvements offered by our model.

7.2.1 Overview of the framework

In the context of language model-based DTI prediction frameworks, the protein embeddings produced by protein language models are inherently distinct for each protein sequence, just as the drug embeddings generated by chemical language models remain independent for different drug sequences [238]. Although similar proteins or drugs should generate similar embeddings, enhancements to these embeddings can be achieved by explicitly defining a neighborhood based on similarities or interactions between drugs or proteins. Conversely, in GAT-based DTI prediction frameworks, various encoding methods such as integer encoding, Word2Vec, position-specific scoring matrix, or biological property-based encoding are utilized to prepare the protein sequences.

For drug sequences, encodings like molecular fingerprint, molecular graph, and Word2Vec are employed as input for the GAT model [241, 242, 243, 244, 245]. As a step toward an integrated approach, we propose combining both strategies by encoding the protein and drug sequences using language models and subsequently generating the final representations through the GAT model. Figure 7.1 illustrates the overall workflow of DTI-LM.

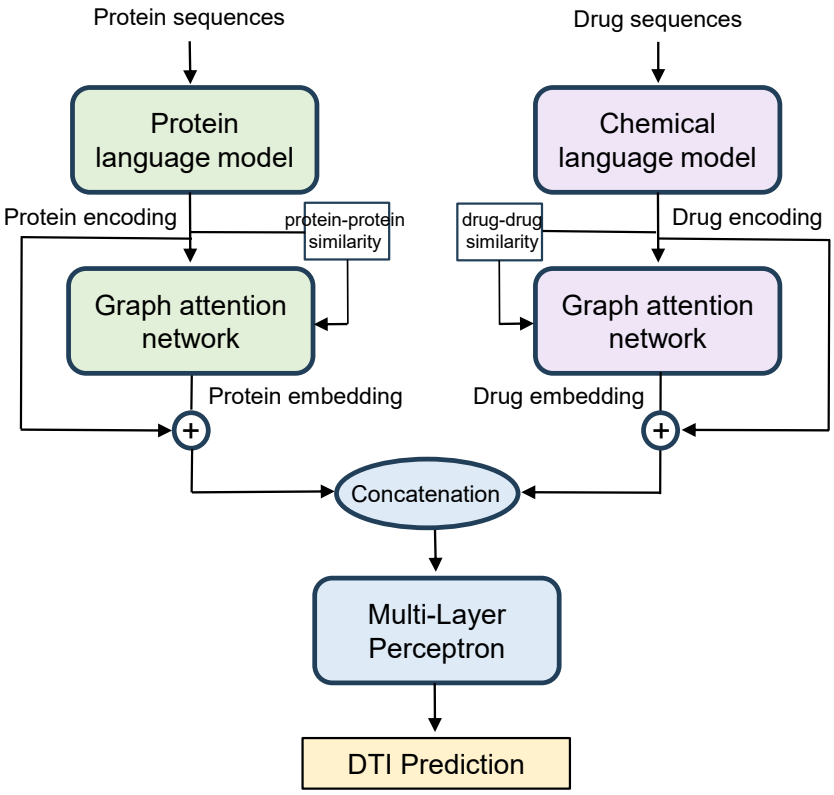

Figure 7.1: Overall framework of DTI-LM. In the framework, protein and drug sequences are fed into their respective language models. Next, the generated encoding and their similarity matrix are used in a graph attention network to generate protein and drug embeddings. The embeddings are then concatenated and passed into a multi-layer perceptron to predict DTI.

The notations used to define the proposed model are summarized in Table 7.1. Let $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_m]$ represent the p-dimensional encodings for m proteins generated by the protein language model

from protein sequences represented by amino acids, where $\boldsymbol{x}_i$ denotes the i^{th} protein. Similarly, $\boldsymbol{Y} = [\boldsymbol{y}_1, \boldsymbol{y}_2, ..., \boldsymbol{y}_n]$ represents the q-dimensional encodings for n drugs generated from drug SMILES sequences. $\boldsymbol{Z}_x$ and $\boldsymbol{Z}_y$ are GAT protein and drug embeddings, respectively, where k, l, and h represent the protein embedding size, drug embedding size, and the number of heads in the GAT. The proposed framework is designed for binary prediction of the drug-target interaction matrix, denoted by $\boldsymbol{I}$. For the remainder of the manuscript, outputs from the LMs are designated as encodings, and outputs from the GATs are designated as embeddings to easily differentiate between them.

7.2.1.1 Protein encoding

We use ESM-2 [231], a 33-layer, 650-million-parameters model with an output dimension of 1280 for encoding protein sequences. It is an advanced deep learning model specifically designed to capture the complex evolutionary patterns and structural features embedded within protein sequences.

Table 7.1: Notations used in DTI-LM

Name	Definition
p, q, m, n, k, l, h	protein encoding size, drug encoding size, number of proteins, number of drugs, protein GAT embedding size, drug GAT embedding size, number of heads respectively
$\boldsymbol{X} \in \mathbb{R}^{p \times m}$	protein sequence encoding generated by ESM-2
$\boldsymbol{Y} \in \mathbb{R}^{q \times n}$	drug SMILES encoding generated by ChemBERTa
$\boldsymbol{S}_x \in \mathbb{R}^{m \times m}$	protein-protein adjacency matrix
$\boldsymbol{S}_y \in \mathbb{R}^{n \times n}$	drug-drug adjacency matrix
$\boldsymbol{Z}_x \in \mathbb{R}^{kh \times m}$	protein embeddings generated by GAT
$\boldsymbol{Z}_y \in \mathbb{R}^{lh \times n}$	drug embeddings generated by GAT
$\boldsymbol{I} \in \mathbb{R}^{m \times n}$	drug-target interaction matrix

The model is trained on the UniRef50 dataset, which is part of the UniProt Knowledgebase [246], a centralized repository for protein sequences and functional information. The dataset is constructed through the clustering of UniRef90 seed sequences, ensuring that each cluster comprises sequences with a minimum of 50% sequence identity to, and 80% overlap with, the longest sequence in the cluster and consists of 11,862,245 clusters [247]. By encoding protein sequences using ESM-2, we can harness the model's capacity to capture long-range dependencies and subtle sequence motifs, thereby facilitating more accurate predictions of protein properties, functions, and interactions.

7.2.1.2 Drug encoding

For drug SMILES sequence encoding, we choose a prominent chemical language model, ChemBERTa [234], a 6-attention layer, 84-million-parameters model with an output dimension of 768. It was trained on 10 million SMILES sequences from the PubChem database [248]. ChemBERTa integrates the powerful language understanding capabilities of BERT with domain-specific knowledge from the chemical and pharmaceutical realms. By encoding drug SMILES sequences, ChemBERTa enables the extraction of rich semantic representations, capturing intricate molecular structures, functional groups, and chemical properties embedded within the SMILES notations. With its capacity to comprehend complex chemical structures and their relationships, ChemBERTa serves as a valuable tool for drug discovery. In this study, we implemented our model using the Hugging Face library [249], a widely recognized and extensively utilized platform for natural language processing and deep learning research.

7.2.1.3 Drug-target interaction prediction

Protein and drug encodings, given by $\boldsymbol{X}$ and $\boldsymbol{Y}$ respectively, are fed into two GATs to derive embeddings by integrating neighborhood information. To define the neighborhood of a protein, an

$m \times m$ Pearson correlation matrix $\boldsymbol{S}_x$ is first calculated. This correlation-based similarity matrix is then converted into a binary adjacency matrix using a threshold where high correlation scores above that threshold are assigned value of 1 while low scores below that threshold are assigned value of 0. The binarized adjacency matrix will be later used to mask the attention coefficients of the model. Whether to keep self-connections in the adjacency matrix and the thresholds used for binarization are set as hyperparameters in the framework and tuned for the best performance. A similar process is applied to obtain the drug neighborhood $\boldsymbol{S}_y$. The model can accommodate other neighborhood definitions such as the protein-protein interaction network (PPI) and drug-drug interaction network (DDI). Once we have the adjacency matrices, we can generate the embeddings for $\boldsymbol{X}$ and $\boldsymbol{Y}$. For protein embedding, the attention directed to $\boldsymbol{x}_i$ from its neighbor $\boldsymbol{x}_j$ can be computed as follows:

$$c_{ij} = \boldsymbol{a}[\boldsymbol{W}\boldsymbol{x}_i || \boldsymbol{W}\boldsymbol{x}_j] \tag{7.1}$$

where $\boldsymbol{W} \in \mathbb{R}^{k \times p}$ and $\boldsymbol{a} \in \mathbb{R}^{1 \times 2k}$ represent the learnable weight parameters of a single head. Here, k denotes the embedding size of the GAT, and $||$ denotes the concatenation operation. Subsequently, the calculated attention values undergo a *LeakyReLU* activation function. To incorporate the structural information of the network, the attention values are modified by applying a mask using the adjacency matrix. Specifically, only the attention values corresponding to connected nodes in the adjacency matrix $\boldsymbol{S}_x$ are retained, while all other values are set to zero. The attention coefficient for a neighbor $\boldsymbol{x}_j$ is then calculated using the *Softmax* function as follows:

$$\alpha_{ij} = \frac{\exp(\textit{LeakyReLU}(\boldsymbol{c}_{ij}))}{\sum_{r \in \mathcal{N}_i} \exp(\textit{LeakyReLU}(\boldsymbol{c}_{ir}))} \tag{7.2}$$

where $\mathcal{N}_i$ represents the neighborhood of the i^{th} protein. The embedding of $\boldsymbol{x}_i$ is calculated as:

$$\boldsymbol{x}'_i = \sigma(\sum_{j \in \mathcal{N}_i} \alpha_{ij} \boldsymbol{W}\boldsymbol{x}_j) \tag{7.3}$$

where σ is a non-linear activation function. We employ multi-head attention mechanism to capture complex relationships and enhance the expressiveness of the learned representations. For h number of heads, each with its separate attention mechanism, the final embedding of the sample is obtained by concatenating the output of the heads. Therefore, the final embedding of the i^{th} protein is given by:

$$\boldsymbol{z}_i = \Big|\Big|_{\mathfrak{h}=1}^{h} \sigma(\sum_{j \in \mathcal{N}_i} \alpha_{ij}^{\mathfrak{h}} \boldsymbol{W}^{\mathfrak{h}} \boldsymbol{x}_j). \tag{7.4}$$

We obtain the embeddings for all m proteins as $\boldsymbol{Z}_x \in \mathbb{R}^{kh \times m}$ and follow the same procedure to obtain the embeddings for n drugs as $\boldsymbol{Z}_y \in \mathbb{R}^{lh \times n}$, where l is the embedding size for drugs from a single head. We design the GAT model to have the same embedding size as LM encoding i.e. $kh = p$ and $lh = q$. For simplicity, we show same number of heads h for drugs and proteins which can be different in implementation of DTI-LM. The number of heads and number of layers in the networks used for generating protein and drug embeddings are set as hyperparameters in the model.

Finally, the protein embedding $\boldsymbol{Z}_x$ and the encoding from the language model $\boldsymbol{X}$ are added together to obtain the final protein representations. Similarly, the drug embedding $\boldsymbol{Z}_y$ and the encoding from the language model $\boldsymbol{Y}$ are added together to obtain the final drug representations. These representations are concatenated and fed into a multilayer perceptron (MLP) to predict the corresponding interactions, as given by:

$$\tilde{\boldsymbol{I}} = MLP([\boldsymbol{Z}_x + \beta \boldsymbol{X}]||[\boldsymbol{Z}_y + \gamma \boldsymbol{Y}]) \tag{7.5}$$

β and γ are hyperparameters that control the contribution of the residual connection. The model is trained with binary cross-entropy loss, calculated as:

$$\mathcal{L} = -\frac{1}{mn} \sum_{i=0}^{mn} [I_i \cdot \log \sigma(\tilde{I}_i) + (1 - I_i) \cdot \log(\sigma(1 - \tilde{I}_i))] \tag{7.6}$$

where σ represents the *Sigmoid* function.

7.2.2 Baselines models

We employ several baselines to compare the performance of our proposed model, DTI-LM. DeepDTA [52], DeepDTI [53], and TransDTI [238] are end-to-end models that take protein and drug sequences as input, similar to DTI-LM. DeepDTA and DeepDTI use convolutional neural networks and deep belief networks, respectively, to process the protein and drug sequences. TransDTI, on the other hand, uses language models for protein and drug sequences with an MLP on top of the outputs from the language models. Additionally, DTI-LM is compared against heterogeneous data-driven models such as DTiGEMS+ [42], DTINet [41], KGE_NFM [1], and TriModel [250] that require more data modalities to train than DTI-LM. Although DTI-LM uses protein-protein and drug-drug similarity matrices, we can generate these matrices from the language model encoding without any external information.

7.3 Experiments

7.3.1 Dataset

The proposed framework is evaluated on four datasets: DrugBank [251], BindingDB [252], Yamanishi_08 [253], and Luo's dataset [41]. The DrugBank and BindingDB datasets contain only protein and drug sequences; therefore, they were primarily utilized for comparing sequence-based methods. In contrast, the Yamanishi_08 and Luo's datasets include heterogeneous knowledge graphs (KG) alongside protein and drug sequences, making them suitable for comparing both sequence-based and heterogeneous data-driven methods. The Yamanish_08 network encompasses 25,487 nodes and 95,579 edges, whereas Luo's dataset network consists of 12,015 nodes and

Table 7.2: Data statistics.

Dataset	Proteins	Drugs	KG	Interactions
DrugBank	2203	1603	No	6041
BindingDB	879	9144	No	4040
Yamanishi_08	722	791	Yes	3448
Luo's	1129	708	Yes	1526

1,895,445 edges. Statistics of the datasets can be found in Table 7.2.

7.3.2 Running DTI-LM

First, the DrugBank and BindingDB datasets are split into training, validation, and test sets, with ratios of 0.79, 0.01, and 0.20, respectively. This splitting process adheres to three specific conditions: warm start (the same drugs and proteins being allowed in both training and test sets), cold start for drugs (drugs in training and test sets are exclusive), and cold start for proteins (proteins in training and test sets are exclusive). The Yamanishi_08 and Luo's datasets are obtained from the source mentioned in [1], and the same training and test splits as utilized in that study are employed to generate our results. While sequence-based models, including DTI-LM, are exclusively trained on the sequences, heterogeneous data-driven models incorporate the use of KG as well. Therefore, heterogeneous data-driven models are not compared on DrugBank and BindingDB datasets. DrugBank, Yamanishi_08, and Luo's datasets provide binary interaction details that were used in our classification framework to train a binary classifier to predict interaction or no interaction for a pair of drug and protein. In contrast, BindingDB provides binding affinity (Kd) data, which is converted into a binary format using a threshold to align with the classification framework. The threshold is chosen to maintain a comparable DTI density as other datasets. The hyperparameters

of the framework are fine-tuned using Ray Tune [254], and comprehensive information regarding the selection of hyperparameters can be found in the Supplementary Document (Table S4). All predictions are run 10 times with different splittings, with the mean area under the Receiver Operating Characteristic curve (AUROC) and the area under the Precision-Recall curve (AUPRC) reported in the respective tables. These experiments are repeated with two variations in the ratios of positive and negative samples in the datasets: balanced data has a 1:1 ratio, whereas unbalanced data has a 1:10 ratio between positive and negative drug-target pairs or all samples if the ratio is less than 1:10.

DTI-LM is thoroughly evaluated through various experiments. Firstly, we compare the performance of DTI prediction with cutting-edge baselines, highlighting the improvements introduced by our model. Subsequently, we conduct an in-depth analysis of DTI-LM to examine its benefits and drawbacks, specifically focusing on the use of the language model-based encoding for DTI prediction.

7.3.3 Prediction results

We designed two DTI prediction scenarios to illustrate the ability of DTI-LM. Firstly, we conducted a comparative analysis of our model against other sequence-based models using DrugBank and BindingDB datasets, demonstrating the enhanced predictive capabilities of our approach relying solely on sequence data. We repeated the experiments with all three types of splitting, each with balanced and unbalanced datasets. Secondly, we pitted our model against heterogeneous data-driven models using Yamanishi_08 and Luo's datasets, highlighting our competitive performance despite utilizing only a fraction of the input data. Not only is protein and drug sequence data more readily available, but it can also significantly reduce the computational complexity of a model compared to heterogeneous data-driven models. In Tables 7.3, 7.4, 7.5, and 7.6, the first row associated

Table 7.3: The classification performance on DrugBank dataset. Average AUROC and AUPRC scores of drug-target prediction for warm start, cold start for drug, and cold start for protein data splitting.

		DTI-LM	TransDTI	DeepDTA	DeepDTI
balanced	warm start	**0.951**	0.934	0.889	0.916
		0.953	0.935	0.882	0.914
	cold start for drug	**0.902**	0.877	0.874	0.859
		0.899	0.889	0.871	0.868
	cold start for protein	**0.923**	0.916	0.855	0.838
		0.935	0.920	0.825	0.850
unbalanced	warm start	**0.960**	0.952	0.907	0.947
		0.863	0.858	0.623	0.773
	cold start for drug	**0.890**	0.876	0.765	0.860
		0.674	0.651	0.441	0.582
	cold start for protein	**0.938**	0.916	0.737	0.871
		0.821	0.789	0.441	0.614

with each splitting strategy represents the AUROC, while the second row depicts the AUPRC.

The results presented in Table 7.3 and Table 7.4 showcase the average classification results of the sequence-based model applied to the DrugBank and BindingDB datasets, respectively. They highlight that our model outperformed the baseline models in the majority of cases. Notably, under the warm start scenario, our model consistently demonstrated superior performance compared to all the baselines across both datasets. The most substantial performance enhancement was observed in the case of cold start for protein splitting despite doing worse than DeepDTA in unbalanced BindingDB dataset. Across different splitting scenarios, our model exhibited an average improvement in AUROC of 3.57% and AUPRC of 8.33% for warm start, 3.84% and 6.13% for cold start for drug, and 5.57% and 8.93% for cold start for protein predictions, respectively. AUROC scores are better in unbalanced splittings due to higher volume of training data. AUPRC scores are un-

Table 7.4: The classification performance on BindingDB dataset. Average AUROC and AUPRC scores of drug-target prediction for warm start, cold start for drug, and cold start for protein data splitting.

		DTI-LM	TransDTI	DeepDTA	DeepDTI
balanced	warm start	**0.939**	0.926	0.868	0.923
		0.934	0.918	0.729	0.910
	cold start for drug	**0.872**	0.870	0.754	0.863
		0.879	0.878	0.699	**0.886**
	cold start for protein	**0.812**	0.809	0.697	0.757
		0.787	0.779	0.572	0.767
unbalanced	warm start	**0.945**	0.941	0.820	0.935
		0.839	0.834	0.577	0.813
	cold start for drug	0.895	0.872	0.851	**0.896**
		0.744	0.708	0.637	0.743
	cold start for protein	0.831	0.818	**0.869**	0.761
		0.463	0.456	**0.568**	0.366

surprisingly lower for unbalanced splittings as there are far less positive interactions compared to negative interactions that makes positive interaction predictions more challenging. We also find that DeepDTA is more unstable compared to other models with a large gap of performance between balanced and unbalanced splitting. It works better for balanced data in DrugBank while doing better for unbalanced data in BindingDB.

Next, Tables 7.5 and 7.6 report the average classification results for both sequence-based and heterogeneous data-driven models on Yamanishi_08 and Luo's datasets. Using the same publicly available data splits as [1] enables a direct comparison of our results with those reported in that paper. As observed, heterogeneous data-driven baselines DTiGEMS+, DTINet, TriModel, and KGE_NFM consistently outperform sequence-based baselines DeepDTI and MPNN_CNN across various scenarios, with a notable performance gap for cold start for drug and cold start for protein

Table 7.5: The classification performance on Yamanishi_08 dataset. Average AUROC and AUPRC scores of drug-target prediction for warm start, cold start for drug, and cold start for protein data splitting. DeepDTI, MPNN_CNN, DTiGEMS+, TriModel, and KGE_NFM results are directly reproduced from [1].

		sequences-based				heterogeneous data-driven		
		DTI-LM	TransDTI	DeepDTI	MPNN_CNN	DTiGEMS+	TriModel	KGE_NFM
balanced	warm start	**0.974**	0.969	0.865	0.834	0.964	0.951	0.968
		0.966	0.961	0.820	0.788	0.957	0.946	0.961
unbalanced	warm start	0.984	0.984	0.982	0.974	0.976	**0.985**	0.983
		0.930	0.927	0.917	0.874	0.874	0.886	0.902
	cold start for drug	0.785	0.762	0.628	0.629	0.745	0.817	**0.853**
		0.451	0.442	0.191	0.194	0.518	0.503	**0.521**
	cold start for protein	0.911	0.902	0.497	0.502	0.674	0.829	**0.921**
		0.739	0.729	0.099	0.098	0.443	0.483	0.679

splittings. Despite being a sequence-based model, DTI-LM not only outperforms other sequence-based baselines but also surpasses heterogeneous data-driven models for warm start and cold start for protein prediction. For cold start for drug splitting, while we outperform other sequence-based baselines in most cases, except MPNN_CNN on Luo's dataset, we still lag behind state-of-the-art heterogeneous data-driven models. This underscores the findings from Tables 7.3 and 7.4 that DTI-LM is more effective for cold start for protein splitting than cold start for drug splitting. To gain a deeper understanding of the factors contributing to the superior performance of our model in the context of cold start for protein as opposed to cold start for drug, we conducted an investigation detailed in section 7.3.5.

Table 7.6: The classification performance on Luo's dataset. Average AUROC and AUPRC scores of drug-target prediction for warm start, cold start for drug, and cold start for protein data splitting. DeepDTI, MPNN_CNN, DTINet, and KGE_NFM results are directly reproduced from [1].

		sequences-based				heterogeneous data-driven	
		DTI-LM	TransDTI	DeepDTI	MPNN_CNN	DTINet	KGE_NFM
balanced	warm start	**0.944**	0.938	0.859	0.830	0.940	0.903
		0.948	0.939	0.840	0.805	0.941	0.898
unbalanced	warm start	**0.971**	**0.971**	0.952	0.929	0.944	0.962
		0.906	0.902	0.793	0.705	0.817	0.855
	cold start	0.760	0.742	0.662	0.806	0.853	**0.881**
	for drug	0.393	0.383	0.225	0.462	**0.592**	0.555
	cold start	**0.832**	0.823	0.487	0.431	0.778	0.813
	for protein	**0.595**	0.589	0.092	0.078	0.388	0.444

7.3.4 Transition from cold start to warm start

Given the limitations in cold start for drug splitting, we investigated the transition between a cold start and warm start prediction to determine the minimum information needed for the transition. For each drug in the test set, we sent a number of samples (drug-target pair) to the training set and tracked how the prediction performance changes with the inclusion of additional information. All predictions with leaked data are also computed 10 times similar to previous results. Figure 7.2 illustrates the results for the DrugBank dataset, where we leaked two, four, and six samples from each drug in the test set to the training set but kept at least one sample for those drugs in the test set. AUPRC has a larger gap between warm start and cold start scenario compared to AUROC. The figure shows that, AUPRC jumps significantly with inclusion of just 2 samples on average for each test drug that is comparable to warm start predictions. Both AUROC and AUPRC keep gradually increasing as we leak more samples.

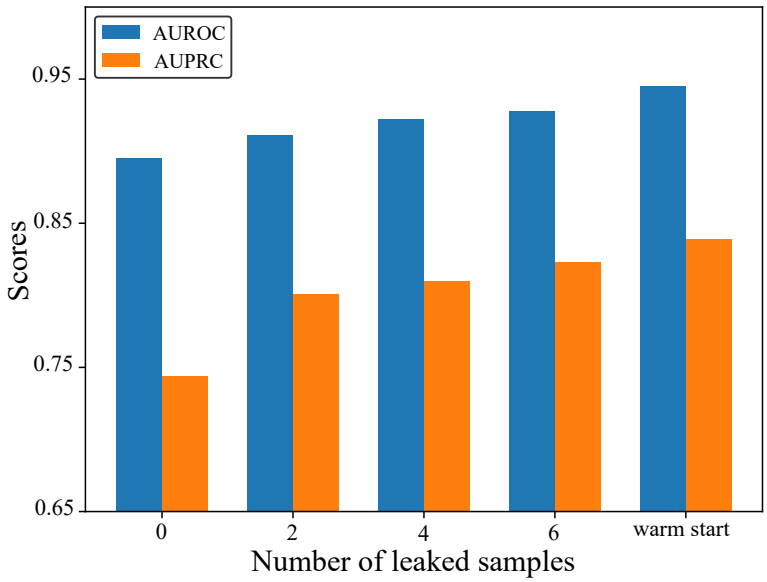

Figure 7.2: Effect of leaked samples. AUROC and AUPRC scores after 2, 4, and 6 samples leaked into training of cold start for drug prediction.

7.3.5 Language model encoding analysis

In this section, we examine the current strengths and weaknesses of language model-based DTI prediction. As observed in the results reported above, DTI-LM performs better in warm start and cold start for protein predictions but lags behind in cold starts for drug predictions. In contrast, other 1D sequence-based methods struggle with both cold starts for protein and cold starts for drug predictions. For cold start predictions, performance depends on how much the model can learn about an unknown drug or protein from the known drugs or proteins in the training data. The results suggest that DTI-LM effectively learned representations for unknown proteins, given the high AUROC and AUPRC values in cold starts for protein prediction. However, it fails to replicate a similar level of learning for unknown drugs. If the representations are significantly different in the training and test sets for a pair of drugs that share similar interactions, this difference can explain

the poor performance in cold starts for drug prediction. Therefore, we compute the similarity of drugs and proteins using their respective SMILES and amino acid sequences, as well as the encoding generated by language models, to inspect the efficiency of the language models in finding similar drugs and proteins.

Table 7.7: Sequence and encoding similarity. Similarity is measured based on the raw sequences and language model encodings representing drugs and proteins.

	raw sequences		LM encoding	
dataset	drug	protein	drug	protein
DrugBank	0.101	0.072	0.644	0.853
BindingDB	0.117	0.090	0.574	0.859
Yamansihi_08	0.104	0.089	0.554	0.853
Luo's dataset	0.097	0.078	0.488	0.845

Table 7.8: Top 5 neighbor support. Average percentage of interactions shared by majority of the neighbors.

	raw sequences		LM encoding	
dataset	drug	protein	drug	protein
DrugBank	25.1%	0.0%	14.3%	30.7%
Yamansihi_08	14.1%	21.5%	6.5%	44.0%
Luo's dataset	30.9%	0.0%	24.5%	26.4%

Table 7.7 shows the similarity of drugs and proteins in the benchmark datasets. For drug similarity using SMILES sequences, we utilize the RDKit library [255] to measure Tanimoto similarity on Morgan fingerprints. Clustal Omega [256] is employed to determine amino acid sequence similarity for proteins. On the other hand, for language model encoding similarities, we calculate the Pearson correlation for each pair of drugs or proteins separately, based on the representations generated by the language models. This process generates two $m \times m$ protein-protein similarity

matrices and two $n \times n$ drug-drug similarity matrices. The mean similarity for all drug/protein pairs is reported in Table 7.7. As shown, neither drug nor protein sequences exhibit significant similarity. It's important to note that sequence-level drug and protein similarity is not directly comparable. However, both similarity metrics have a range of 0-1, with 1 indicating the highest similarity. The lack of significant similarity is evident. In contrast, the language model encodings are highly similar across all datasets, particularly in the case of protein encoding. This underscores the greater ability of the protein language model (ESM-2) to capture protein similarity even when amino acid sequences are not very similar. However, it remains a possibility that ESM-2 generates all protein encodings similarly, regardless of the actual similarity between them, which may impede DTI prediction. Therefore, we conduct another experiment to investigate whether similar drugs or proteins in the encoding domain also share similar interactions. We measure how many drug-protein interactions of a given drug (or protein) are supported by the majority of its neighboring drugs (or proteins). Neighbors are defined as the top $\mathcal{N}$ similar drugs (proteins) to a drug (protein) using raw sequence or encoding-based similarity matrices. In this experiment, we set $\mathcal{N}$ = 5, and a protein (drug) interaction of a given drug (protein) must be shared by at least 3 of its neighboring drugs (proteins).

Table 7.8 presents the average percentage of interactions supported by the majority (3 or more) of neighbors for a drug or protein. We employ both raw sequence-based similarities and encoding-based similarities to construct the neighborhood. From the table, we can see that drugs receive a higher percentage of support from neighbors compared to proteins when neighbors are selected based on raw sequence-based similarity. However, the average percentage of support for drugs decreases across all datasets when neighbors are selected based on language model encoding. This suggests that encoding similarity in drugs is less meaningful, as similar drugs may exhibit drastically different interactions.

Table 7.8 also illustrates the noteworthy increase in average percentage of support for proteins us-

ing similarity matrix generated from language model encoding compared to raw sequence. For example, 44% of all drug-protein interactions from proteins in Yamanishi_08 dataset are also shared by at least three of their respective neighbor proteins. The presence of a strong neighborhood led us to use GAT to incorporate this vital information in the DTI prediction and our implementation of GAT successfully improves the prediction performance over TransDTI. In light of these findings, we can see why DTI-LM demonstrates substantial improvements in cold start for proteins predictions but faces challenges in the case of drugs. Existing chemical language models may struggle to capture the complex interwoven information in the SMILES sequences as efficiently as ESM-2 does for protein sequence.

7.4 Discussion

In our comprehensive experiments, DTI-LM shows great prediction results, especially for warm start and cold start for proteins scenarios. It successfully overcomes the traditional challenges faced by sequence-based models for cold start for protein prediction. However, it falls short of achieving a comparable level of performance for cold start for drugs, despite improvements over the existing sequence-based models. We delved deeply into analyzing the reasons for the discrepancies between cold start for protein and drug predictions. Our experiments, detailed in Section 7.3.5, show that the ESM-2 is very effective in finding similar proteins that also share similar drug interactions based solely on amino acid sequences. In contrast, ChemBERTa lacks the same level of proficiency for drugs. We also explored the performance of newer, larger models such as ChemGPT [235] and observed similar outcomes.

The experiment outlined in Section 7.3.5 is not conclusive; instead, it gives us a general idea about the performance of the protein and chemical language models. A few crucial aspects of the experiment are discussed below.

- In Table 7.7, we present the Pearson correlation, which ignores the non-linear relationship that can be captured by the subsequent GAT and MLP we employ for the prediction.

- The average neighbor support, as shown in Table 7.8, paints an important but incomplete picture. The training process involves contributions from samples beyond the top 5 neighbors, impacting results irrespective of the quality of these neighbors.

- Finding support for protein interaction and drug interaction may also pose varying levels of difficulty due to the different numbers of drugs and proteins in each dataset. For instance, datasets like DrugBank and Luo's exhibit a lower number of proteins than drugs, i.e., proteins have fewer options to choose from to find an interaction than drugs. Therefore, the probability of proteins sharing similar interactions will be higher than drugs sharing similar interactions. This circumstance can make it comparatively easier to find neighbor proteins with similar drug interactions than neighbor drugs with similar protein interactions. However, Yamanishi_08 has more drugs than proteins (as indicated in Table 7.2) while having the largest difference between support for proteins and drugs, as seen in Table 7.8. Therefore, the difference cannot be completely explained by the number of proteins or drugs.

- It is possible that drugs with similar sequences inherently do not share similar interactions. This makes finding drugs with similar interactions based solely on sequences more challenging. However, we use the support for drug interactions based on raw sequences as a baseline (Table 7.8) and expect the language models to capture more complex similarities. We observe that ESM-2 aligns with this expectation, showing an improved percentage of support in LM encoding compared to raw sequences. On the other hand, ChemBERTa fails to meet the expectation and demonstrates lower support for LM encoding compared to raw sequences. This could be interpreted as similar drug LM encodings being further away from sharing similar interactions than similar SMILES sequences.

The domain of pre-trained language models is improving at an unprecedented level, giving us hope for stronger and more advanced chemical language models in the future. This progress is expected to address cold start for drugs issues more effectively, as ESM-2 has done for cold start for protein predictions.

Based on the higher percentage of support for drugs using raw sequences in Table 7.8, we utilized a raw sequence-based similarity matrix in drug GAT for DTI prediction and found worse results (results are not shown in the manuscript). This can be attributed to the fact that similar SMILES sequences can have different LM encodings; thus, the raw sequence-based neighborhood will be less meaningful for LM encoding. These limitations might be prevalent in all language model-based DTI prediction frameworks that use drug sequence data.

7.5 Conclusion

We propose DTI-LM, a language model-based DTI prediction framework that incorporates neighborhood information for predictions. Our goal is to achieve state-of-the-art results in various prediction scenarios and to test the limits of existing protein and chemical language models for these tasks. DTI-LM outperformed the baselines for warm start and cold start for protein predictions. We also tracked back on the weak performance of DTI-LM for cold start for drug predictions and identified the chemical language model as a limiting factor. Recent notable advancements in natural language processing may pave the way for the development of improved protein and chemical language models to address the cold start problem more efficiently. Nevertheless, DTI-LM currently excels in cold start for protein predictions, a crucial aspect for personalized medicine where tailoring treatment to individual patients' protein variants is essential.

CHAPTER 8: CONCLUSION AND FUTURE WORK

This dissertation delves into the utilization of machine learning algorithms for the analysis of multi-modal data in computational biology. It introduces two integrative models crafted to effectively handle multi-modal data, emphasizing the utilization of inter-modal interaction networks. The dissertation unfolds through structured chapters, encompassing the development and implementation of these models, alongside methodologies devised to surmount common challenges encountered in multi-modal integrative models. In the initial chapters, we present multi-modal integrative models, showcasing their enhanced performance in disease outcome prediction and biomarker identification. Notably, we observe that the efficacy of these models is contingent upon the quality of input data. Given that missing values are indicative of data quality issues, particularly prevalent in biological datasets, we propose two missing value imputation frameworks in subsequent chapters. These frameworks, detailed in the following sections, hold the potential to significantly enhance the accuracy and feasibility of downstream predictions. Lastly, we introduce a link prediction framework aimed at mitigating noise within the interaction network. Given the reliance of our proposed multi-modal integrative models on the accuracy of the interaction network, the link prediction model facilitates a smoother flow of information across data modalities.

In the future, our intention is to expand the capabilities of the multi-modal integrative models to encompass the integration of more than two data modalities. Physiological activities are intricately influenced by the interactions among numerous omics layers. Many lingering questions in human biology stem from our limited understanding of this multi-layered structure and our inability to comprehensively explore it. Integrating all omics data and their respective interaction networks holds promise in elucidating the underlying disease mechanisms with greater precision. Furthermore, we aim to amalgamate the missing value imputation and interaction network methodologies with the integrative models to enhance input data quality and diminish noise in a unified approach.

Alternative generative models, such as diffusion models, could be explored as replacements for GANs to merge multi-omics datasets. Additionally, there is potential to incorporate more domain-specific knowledge or biological constraints to bolster prediction accuracy and reduce noise.

Validation of the proposed methodologies and models on a broader array of real-world biological datasets is on our agenda, collaborating closely with domain experts to evaluate their efficacy in specific computational biology tasks. We also seek to address scalability and efficiency concerns inherent in the proposed frameworks, particularly when handling large-scale multi-modal datasets, through the exploration of parallel computing techniques and algorithmic optimizations. Lastly, we are committed to enhancing the biological interpretability of the results derived from integrative models and interaction predictions. It is paramount that insights gleaned from computational analyses are not only meaningful but also actionable for biologists and researchers in the field, facilitating advancements in our understanding of complex biological processes.

LIST OF REFERENCES

[1] Qing Ye, Chang-Yu Hsieh, Ziyi Yang, Yu Kang, Jiming Chen, Dongsheng Cao, Shibo He, and Tingjun Hou. A unified drug–target interaction prediction framework based on knowledge graph and recommendation system. *Nature communications*, 12(1):6775, 2021.

[2] Yehudit Hasin, Marcus Seldin, and Aldons Lusis. Multi-omics approaches to disease. *Genome biology*, 18(1):1–15, 2017.

[3] Satoshi Takahashi, Masamichi Takahashi, Shota Tanaka, Shunsaku Takayanagi, Hirokazu Takami, Erika Yamazawa, Shohei Nambu, Mototaka Miyake, Kaishi Satomi, Koichi Ichimura, et al. A new era of neuro-oncology research pioneered by multi-omics analysis and machine learning. *Biomolecules*, 11(4):565, 2021.

[4] Paola Leon-Mimila, Jessica Wang, and Adriana Huertas-Vazquez. Relevance of multi-omics studies in cardiovascular diseases. *Frontiers in cardiovascular medicine*, 6:91, 2019.

[5] Michael Olivier, Reto Asmis, Gregory A Hawkins, Timothy D Howard, and Laura A Cox. The need for multi-omics biomarker signatures in precision medicine. *International journal of molecular sciences*, 20(19):4781, 2019.

[6] Indhupriya Subramanian, Srikant Verma, Shiva Kumar, Abhay Jere, and Krishanpal Anamika. Multi-omics data integration, interpretation, and its application. *Bioinformatics and biology insights*, 14:1177932219899051, 2020.

[7] Yunjin Li, Lu Ma, Duojiao Wu, and Geng Chen. Advances in bulk and single-cell multi-omics approaches for systems biology and precision medicine. *Briefings in Bioinformatics*, 22(5):bbab024, 2021.

[8] Bo Wang, Aziz M Mezlini, Feyyaz Demir, Marc Fiume, Zhuowen Tu, Michael Brudno, Benjamin Haibe-Kains, and Anna Goldenberg. Similarity network fusion for aggregating data types on a genomic scale. *Nature methods*, 11(3):333, 2014.

[9] Dong Ouyang, Yong Liang, Le Li, Ning Ai, Shanghui Lu, Mingkun Yu, Xiaoying Liu, and Shengli Xie. Integration of multi-omics data using adaptive graph learning and attention mechanism for patient classification and biomarker identification. *Computers in Biology and Medicine*, 164:107303, 2023.

[10] Adib Shafi, Tin Nguyen, Azam Peyvandipour, Hung Nguyen, and Sorin Draghici. A multi-cohort and multi-omics meta-analysis framework to identify network-based gene signatures. *Frontiers in Genetics*, 10:159, 2019.

[11] Marco Chierici, Nicole Bussola, Alessia Marcolini, Margherita Francescatto, Alessandro Zandonà, Lucia Trastulla, Claudio Agostinelli, Giuseppe Jurman, and Cesare Furlanello. Integrative network fusion: a multi-omics approach in molecular profiling. *Frontiers in oncology*, 10:1065, 2020.

[12] Ying Yang, Sha Tian, Yushan Qiu, Pu Zhao, and Quan Zou. Mdicc: novel method for multi-omics data integration and cancer subtype identification. *Briefings in Bioinformatics*, 23(3):bbac132, 2022.

[13] Qianqian Song, Jing Su, and Wei Zhang. scgcn is a graph convolutional networks algorithm for knowledge transfer in single cell omics. *Nature communications*, 12(1):3826, 2021.

[14] Zhi-Jie Cao and Ge Gao. Multi-omics single-cell data integration and regulatory inference with graph-linked embedding. *Nature Biotechnology*, 40(10):1458–1466, 2022.

[15] Anjun Ma, Xiaoying Wang, Jingxian Li, Cankun Wang, Tong Xiao, Yuntao Liu, Hao Cheng, Juexin Wang, Yang Li, Yuzhou Chang, et al. Single-cell biological network inference using a heterogeneous graph transformer. *Nature Communications*, 14(1):964, 2023.

[16] Paulina Krzyszczyk, Alison Acevedo, Erika J Davidoff, Lauren M Timmins, Ileana Marrero-Berrios, Misaal Patel, Corina White, Christopher Lowe, Joseph J Sherba, Clara Hartmanshenn, et al. The growing role of precision and personalized medicine for cancer treatment. *Technology*, 6(03n04):79–100, 2018.

[17] Chao Wang, Raghu Machiraju, and Kun Huang. Breast cancer patient stratification using a molecular regularized consensus clustering method. *Methods*, 67(3):304–312, 2014.

[18] Khandakar Tanvir Ahmed, Sunho Park, Qibing Jiang, Yunku Yeu, TaeHyun Hwang, and Wei Zhang. Network-based drug sensitivity prediction. *BMC medical genomics*, 13(11):1–10, 2020.

[19] Nimrod Rappoport and Ron Shamir. NEMO: Cancer subtyping by integration of partial multi-omic data. *Bioinformatics*, 35(18):3348–3356, 2019.

[20] Hung Nguyen, Sangam Shrestha, Sorin Draghici, and Tin Nguyen. PINSPlus: a tool for tumor subtype discovery in integrated genomic data. *Bioinformatics*, 35(16):2843–2846, 2019.

[21] Ricard Argelaguet, Britta Velten, Damien Arnol, Sascha Dietrich, Thorsten Zenz, John C Marioni, Florian Buettner, Wolfgang Huber, and Oliver Stegle. Multi-Omics Factor Analysis—a framework for unsupervised integration of multi-omics data sets. *Molecular systems biology*, 14(6):e8124, 2018.

[22] Tao Zhou, Kim-Han Thung, Xiaofeng Zhu, and Dinggang Shen. Effective feature learning and fusion of multimodality data using stage-wise deep neural network for dementia diagnosis. *Human brain mapping*, 40(3):1001–1016, 2019.

[23] Hiromi WL Koh, Damian Fermin, Christine Vogel, Kwok Pui Choi, Rob M Ewing, and Hyungwon Choi. iOmicsPASS: network-based integration of multiomics data for predictive subnetwork discovery. *NPJ systems biology and applications*, 5(1):1–10, 2019.

[24] Xiao Li, Jie Ma, Ling Leng, Mingfei Han, Mansheng Li, Fuchu He, and Yunping Zhu. Mogcn: a multi-omics integration method based on graph convolutional network for cancer subtype analysis. *Frontiers in Genetics*, 13:806842, 2022.

[25] Tongxin Wang, Wei Shao, Zhi Huang, Haixu Tang, Jie Zhang, Zhengming Ding, and Kun Huang. Mogonet integrates multi-omics data using graph convolutional networks allowing patient classification and biomarker identification. *Nature Communications*, 12(1):3445, 2021.

[26] Ziynet Nesibe Kesimoglu and Serdar Bozdag. Supreme: multiomics data integration using graph convolutional networks. *NAR Genomics and Bioinformatics*, 5(2):lqad063, 2023.

[27] Conghao Wang, Wu Lue, Rama Kaalia, Parvin Kumar, and Jagath C Rajapakse. Network-based integration of multi-omics data for clinical outcome prediction in neuroblastoma. *Scientific Reports*, 12(1):15425, 2022.

[28] Luciana O Silva and Luis E Zárate. A brief review of the main approaches for treatment of missing data. *Intelligent Data Analysis*, 18(6):1177–1198, 2014.

[29] Abinash Sahoo and Dillip Kumar Ghose. Imputation of missing precipitation data using KNN, SOM, RF, and FNN. *Soft Computing*, pages 1–18, 2022.

[30] Hsiang-Fu Yu, Nikhil Rao, and Inderjit S Dhillon. Temporal regularized matrix factorization for high-dimensional time series prediction. *Advances in neural information processing systems*, 29, 2016.

[31] Alexandre Hippert-Ferrer, Yajing Yan, and Philippe Bolon. EM-EOF: Gap-filling in incomplete SAR displacement time series. *IEEE Transactions on Geoscience and Remote Sensing*, 59(7):5794–5811, 2020.

[32] Jinsung Yoon, William R Zame, and Mihaela van der Schaar. Estimating missing data in temporal data streams using multi-directional recurrent neural networks. *IEEE Transactions on Biomedical Engineering*, 66(5):1477–1490, 2018.

[33] Vincent Fortuin, Dmitry Baranchuk, Gunnar Rätsch, and Stephan Mandt. Gp-vae: Deep probabilistic time series imputation. In *International conference on artificial intelligence and statistics*, pages 1651–1661. PMLR, 2020.

[34] Xiaoye Miao, Yangyang Wu, Jun Wang, Yunjun Gao, Xudong Mao, and Jianwei Yin. Generative semi-supervised learning for multivariate time series imputation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 8983–8991, 2021.

[35] Khandakar Tanvir Ahmed, Sze Cheng, Qian Li, Jeongsik Yong, and Wei Zhang. Incomplete time-series gene expression in integrative study for islet autoimmunity prediction. *Briefings in Bioinformatics*, 2022.

[36] Yue Bai, Lichen Wang, Zhiqiang Tao, Sheng Li, and Yun Fu. Correlative channel-aware fusion for multi-view time series classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 6714–6722, 2021.

[37] Khandakar Tanvir Ahmed, Jiao Sun, William Chen, Irene Martinez, Sze Cheng, Wencai Zhang, Jeongsik Yong, and Wei Zhang. In silico model for mirna-mediated regulatory network in cancer. *Briefings in Bioinformatics*, 2021.

[38] Khandakar Tanvir Ahmed, Jiao Sun, Sze Cheng, Jeongsik Yong, and Wei Zhang. Multi-omics data integration by generative adversarial network. *Bioinformatics*, 38(1):179–186, 2022.

[39] Xiang Zhou, Hua Chai, Huiying Zhao, Ching-Hsing Luo, and Yuedong Yang. Imputing missing RNA-seq data from DNA methylation by using transfer learning based neural network. *bioRxiv*, page 803692, 2020.

[40] TEDDY Study Group et al. The environmental determinants of diabetes in the young (TEDDY) study. *Annals of the New York Academy of Sciences*, 1150:1, 2008.

[41] Yunan Luo, Xinbin Zhao, Jingtian Zhou, Jinglin Yang, Yanqing Zhang, Wenhua Kuang, Jian Peng, Ligong Chen, and Jianyang Zeng. A network integration approach for drug-target interaction prediction and computational drug repositioning from heterogeneous information. *Nature communications*, 8(1):573, 2017.

[42] Maha A Thafar, Rawan S Olayan, Haitham Ashoor, Somayah Albaradei, Vladimir B Bajic, Xin Gao, Takashi Gojobori, and Magbubah Essack. DTiGEMS+: drug–target interaction prediction using graph embedding, graph mining, and similarity-based techniques. *Journal of Cheminformatics*, 12(1):1–17, 2020.

[43] Ran Zhang, Zhanjie Wang, Xuezhi Wang, Zhen Meng, and Wenjuan Cui. Mhtan-dti: Metapath-based hierarchical transformer and attention network for drug–target interaction prediction. *Briefings in Bioinformatics*, 24(2):bbad079, 2023.

[44] Yang Li, Guanyu Qiao, Keqi Wang, and Guohua Wang. Drug–target interaction predication via multi-channel graph neural networks. *Briefings in Bioinformatics*, 23(1):bbab346, 2022.

[45] Matthew Ragoza, Joshua Hochuli, Elisa Idrobo, Jocelyn Sunseri, and David Ryan Koes. Protein–ligand scoring with convolutional neural networks. *Journal of chemical information and modeling*, 57(4):942–957, 2017.

[46] Izhar Wallach, Michael Dzamba, and Abraham Heifets. AtomNet: a deep convolutional neural network for bioactivity prediction in structure-based drug discovery. *arXiv preprint arXiv:1510.02855*, 2015.

[47] Marta M Stepniewska-Dziubinska, Piotr Zielenkiewicz, and Pawel Siedlecki. Development and evaluation of a deep learning model for protein–ligand binding affinity prediction. *Bioinformatics*, 34(21):3666–3674, 2018.

[48] Kexin Huang, Cao Xiao, Lucas M Glass, and Jimeng Sun. MolTrans: molecular interaction transformer for drug–target interaction prediction. *Bioinformatics*, 37(6):830–836, 2021.

[49] Mehdi Yazdani-Jahromi, Niloofar Yousefi, Aida Tayebi, Elayaraja Kolanthai, Craig J Neal, Sudipta Seal, and Ozlem Ozmen Garibay. Attentionsitedti: an interpretable graph-based model for drug-target interaction prediction using nlp sentence-level relation classification. *Briefings in Bioinformatics*, 23(4):bbac272, 2022.

[50] Shuangjia Zheng, Yongjian Li, Sheng Chen, Jun Xu, and Yuedong Yang. Predicting drug–protein interaction using quasi-visual question answering system. *Nature Machine Intelligence*, 2(2):134–140, 2020.

[51] Fei Li, Ziqiao Zhang, Jihong Guan, and Shuigeng Zhou. Effective drug–target interaction prediction with mutual interaction neural network. *Bioinformatics*, 38(14):3582–3589, 2022.

[52] Hakime Öztürk, Arzucan Özgür, and Elif Ozkirimli. DeepDTA: deep drug–target binding affinity prediction. *Bioinformatics*, 34(17):i821–i829, 2018.

[53] Ming Wen, Zhimin Zhang, Shaoyu Niu, Haozhi Sha, Ruihan Yang, Yonghuan Yun, and Hongmei Lu. Deep-learning-based drug–target interaction prediction. *Journal of proteome research*, 16(4):1401–1409, 2017.

[54] Lifan Chen, Xiaoqin Tan, Dingyan Wang, Feisheng Zhong, Xiaohong Liu, Tianbiao Yang, Xiaomin Luo, Kaixian Chen, Hualiang Jiang, and Mingyue Zheng. TransformerCPI: improving compound–protein interaction prediction by sequence-based deep learning with self-attention mechanism and label reversal experiments. *Bioinformatics*, 36(16):4406–4414, 2020.

[55] Bo-Wei Zhao, Xiao-Rui Su, Peng-Wei Hu, Yu-An Huang, Zhu-Hong You, and Lun Hu. igrldti: an improved graph representation learning method for predicting drug–target interactions over heterogeneous biological information network. *Bioinformatics*, 39(8):btad451, 2023.

[56] Peiliang Zhang, Ziqi Wei, Chao Che, and Bo Jin. Deepmgt-dti: Transformer network incorporating multilayer graph information for drug–target interaction prediction. *Computers in biology and medicine*, 142:105214, 2022.

[57] Sara Goodwin, John D McPherson, and W Richard McCombie. Coming of age: ten years of next-generation sequencing technologies. *Nature Reviews Genetics*, 17(6):333, 2016.

[58] Joerg Martin Buescher and Edward M Driggers. Integration of omics: more than the sum of its parts. *Cancer & metabolism*, 4:1–8, 2016.

[59] Michal Krassowski, Vivek Das, Sangram K Sahu, and Biswapriya B Misra. State of the field in multi-omics research: from computational needs to data mining and sharing. *Frontiers in Genetics*, 11:610798, 2020.

[60] Mohan Babu and Michael Snyder. Multi-omics profiling for health. *Molecular & Cellular Proteomics*, 22(6), 2023.

[61] Hsin-Sung Yeh, Wei Zhang, and Jeongsik Yong. Analyses of alternative polyadenylation: from old school biochemistry to high-throughput technologies. *BMB reports*, 50(4):201, 2017.

[62] Julia K Nussbacher and Gene W Yeo. Systematic discovery of RNA binding proteins that regulate microRNA levels. *Molecular cell*, 69(6):1005–1016, 2018.

[63] Matthias W Hentze, Alfredo Castello, Thomas Schwarzl, and Thomas Preiss. A brave new world of RNA-binding proteins. *Nature Reviews Molecular Cell Biology*, 19(5):327, 2018.

[64] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27:2672–2680, 2014.

[65] Minseon Kim, Ilhwan Oh, and Jaegyoon Ahn. An improved method for prediction of cancer prognosis by network learning. *Genes*, 9(10):478, 2018.

[66] Arsham Ghahramani, Fiona M Watt, and Nicholas M Luscombe. Generative adversarial networks simulate gene expression and predict perturbations in single cells. *BioRxiv*, page 262501, 2018.

[67] Jinhee Park, Hyerin Kim, Jaekwang Kim, and Mookyung Cheon. A practical application of generative adversarial networks for RNA-seq analysis to predict the molecular progress of Alzheimer's disease. *PLoS computational biology*, 16(7):e1008099, 2020.

[68] Yungang Xu, Zhigang Zhang, Lei You, Jiajia Liu, Zhiwei Fan, and Xiaobo Zhou. scIGANs: single-cell RNA-seq imputation using generative adversarial networks. *Nucleic acids research*, 48(15):e85–e85, 2020.

[69] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017.

[70] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 5907–5915, 2017.

[71] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[72] Noah Simon, Jerome Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for Cox's proportional hazards model via coordinate descent. *Journal of statistical software*, 39(5):1, 2011.

[73] Sebastian Pölsterl. scikit-survival: A Library for Time-to-Event Analysis Built on Top of scikit-learn. *Journal of Machine Learning Research*, 21(212):1–6, 2020.

[74] Cameron Davidson-Pilon. lifelines: survival analysis in Python. *Journal of Open Source Software*, 4(40):1317, 2019.

[75] Cancer Genome Atlas Network TCGA et al. Comprehensive molecular portraits of human breast tumours. *Nature*, 490(7418):61, 2012.

[76] Cancer Genome Atlas Research Network TCGA et al. Comprehensive molecular profiling of lung adenocarcinoma. *Nature*, 511(7511):543, 2014.

[77] Cancer Genome Atlas Research Network TCGA et al. Integrated genomic analyses of ovarian carcinoma. *Nature*, 474(7353):609, 2011.

[78] Mary J Goldman, Brian Craft, Mim Hastie, Kristupas Repečka, Fran McDade, Akhil Kamath, Ayan Banerjee, Yunhai Luo, Dave Rogers, Angela N Brooks, et al. Visualizing and interpreting cancer genomics data via the Xena platform. *Nature Biotechnology*, pages 1–4, 2020.

[79] Jianjiong Gao, Bülent Arman Aksoy, Ugur Dogrusoz, Gideon Dresdner, Benjamin Gross, S Onur Sumer, Yichao Sun, Anders Jacobsen, Rileen Sinha, Erik Larsson, et al. Integrative analysis of complex cancer genomics and clinical profiles using the cBioPortal. *Science signaling*, 6(269):pl1–pl1, 2013.

[80] Vikram Agarwal, George W Bell, Jin-Wu Nam, and David P Bartel. Predicting effective microRNA target sites in mammalian mRNAs. *elife*, 4:e05005, 2015.

[81] Zhi-Ping Liu, Canglin Wu, Hongyu Miao, and Hulin Wu. RegNetwork: an integrated database of transcriptional and post-transcriptional regulatory networks in human and mouse. *Database*, 2015, 2015.

[82] Meng Song, Jonathan Greenbaum, Joseph Luttrell IV, Weihua Zhou, Chong Wu, Hui Shen, Ping Gong, Chaoyang Zhang, and Hong-Wen Deng. A review of integrative imputation for multi-omics datasets. *Frontiers in Genetics*, 11, 2020.

[83] Sini Nagpal, Xiaoran Meng, Michael P Epstein, Lam C Tsoi, Matthew Patrick, Greg Gibson, Philip L De Jager, David A Bennett, Aliza P Wingo, and Thomas S Wingo. TIGAR: an improved Bayesian tool for transcriptomic data imputation enhances gene mapping of complex traits. *The American Journal of Human Genetics*, 105(2):258–266, 2019.

[84] Hossein Sharifi-Noghabi, Olga Zolotareva, Colin C Collins, and Martin Ester. MOLI: multi-omics late integration with deep neural networks for drug response prediction. *Bioinformatics*, 35(14):i501–i509, 2019.

[85] John C Marioni, Christopher E Mason, Shrikant M Mane, Matthew Stephens, and Yoav Gilad. RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome research*, 18(9):1509–1517, 2008.

[86] Zhong Wang, Mark Gerstein, and Michael Snyder. RNA-Seq: a revolutionary tool for transcriptomics. *Nature reviews genetics*, 10(1):57–63, 2009.

[87] Rasko Leinonen, Hideaki Sugawara, Martin Shumway, and International Nucleotide Sequence Database Collaboration. The sequence read archive. *Nucleic acids research*, 39(suppl_1):D19–D21, 2010.

[88] Christine Vogel and Edward M Marcotte. Insights into the regulation of protein abundance from proteomic and transcriptomic analyses. *Nature reviews. Genetics*, 13(4):227, 2012.

[89] I Kosti, N Jain, D Aran, AJ Butte, and M Sirota. Cross-tissue Analysis of Gene and Protein Expression in Normal and Cancer Tissues. *Scientific reports*, 6:24799, 2016.

[90] Fredrik Edfors, Frida Danielsson, Björn M Hallström, Lukas Käll, Emma Lundberg, Fredrik Pontén, Björn Forsström, and Mathias Uhlén. Gene-specific correlation of RNA and protein levels in human cells and tissues. *Molecular systems biology*, 12(10):883, 2016.

[91] Mathias Wilhelm, Judith Schlegl, Hannes Hahne, Amin Moghaddas Gholami, Marcus Lieberenz, Mikhail M Savitski, Emanuel Ziegler, Lars Butzmann, Siegfried Gessulat, Harald Marx, et al. Mass-spectrometry-based draft of the human proteome. *Nature*, 509(7502):582–587, 2014.

[92] Ran Elkon, Alejandro P Ugalde, and Reuven Agami. Alternative cleavage and polyadenylation: extent, regulation and function. *Nature Reviews Genetics*, 14(7):496–506, 2013.

[93] Jae-Woong Chang, Wei Zhang, Hsin-Sung Yeh, Ebbing P De Jong, Semo Jun, Kwan-Hyun Kim, Sun S Bae, Kenneth Beckman, Tae Hyun Hwang, Kye-Seong Kim, Kim Do-Hyung, Timothy J. Griffin, Rui Kuang, and Jeongsik Yong. mRNA 3'-UTR shortening is a molecular signature of mTORC1 activation. *Nature communications*, 6:7218, 2015.

[94] Yonit Hoffman, Debora Rosa Bublik, Alejandro P Ugalde, Ran Elkon, Tammy Biniashvili, Reuven Agami, Moshe Oren, and Yitzhak Pilpel. 3'UTR shortening potentiates microRNA-based repression of pro-differentiation genes in proliferating human cells. *PLoS genetics*, 12(2):e1005879, 2016.

[95] Antonio Lembo, Ferdinando Di Cunto, and Paolo Provero. Shortening of 3' UTRs correlates with poor prognosis in breast and lung cancer. *PloS one*, 7(2):e31129, 2012.

[96] Christine Mayr and David P Bartel. Widespread shortening of 3' UTRs by alternative cleavage and polyadenylation activates oncogenes in cancer cells. *Cell*, 138(4):673–684, 2009.

[97] Rickard Sandberg, Joel R Neilson, Arup Sarma, Phillip A Sharp, and Christopher B Burge. Proliferating cells express mRNAs with shortened 39 untranslated regions and fewer microRNA target sites. *Science*, 320(5883):1643–1647, 2008.

[98] Jie Zhu, Zhibao Zheng, Jia Wang, Jinhua Sun, Pan Wang, Xianying Cheng, Lun Fu, Liming Zhang, Zuojun Wang, and Zhaoyun Li. Different miRNA expression profiles between human breast cancer tumors and serum. *Frontiers in genetics*, 5:149, 2014.

[99] Eleni van Schooneveld, Maartje CA Wouters, Ilse Van der Auwera, Dieter J Peeters, Hans Wildiers, Peter A Van Dam, Ignace Vergote, Peter B Vermeulen, Luc Y Dirix, and Steven J

Van Laere. Expression profiling of cancerous and normal breast tissues identifies microRNAs that are differentially expressed in serum from patients with (metastatic) breast cancer and healthy volunteers. *Breast cancer research*, 14(1):R34, 2012.

[100] Marilena V. Iorio, Manuela Ferracin, Chang-Gong Liu, Angelo Veronese, Riccardo Spizzo, Silvia Sabbioni, Eros Magri, Massimo Pedriali, Muller Fabbri, Manuela Campiglio, Sylvie Ménard, Juan P. Palazzo, Anne Rosenberg, Piero Musiani, Stefano Volinia, Italo Nenci, George A. Calin, Patrizia Querzoli, Massimo Negrini, and Carlo M. Croce. MicroRNA gene expression deregulation in human breast cancer. *Cancer research*, 65(16):7065–7070, 2005.

[101] Nerea Matamala, María Teresa Vargas, Ricardo González-Cámpora, Rebeca Miñambres, José Ignacio Arias, Primitiva Menéndez, Eduardo Andrés-León, Gonzalo Gómez-López, Kira Yanowsky, Julio Calvete-Candenas, Lucía Inglada-Pérez, Beatriz Martínez-Delgado, and Javier Benítez. Tumor microRNA expression profiling identifies circulating microRNAs for early breast cancer detection. *Clinical chemistry*, 61(8):1098–1106, 2015.

[102] Ze-Hua Wang and Cong-Jian Xu. Research progress of microRNA in early detection of ovarian cancer. *Chinese medical journal*, 128(24):3363, 2015.

[103] Harsh Dweep and Norbert Gretz. miRWalk2.0: a comprehensive atlas of microRNA-target interactions. *Nature methods*, 12(8):697–697, 2015.

[104] Nathan Wong and Xiaowei Wang. miRDB: an online resource for microRNA target prediction and functional annotations. *Nucleic acids research*, 43(D1):D146–D152, 2014.

[105] Hsi-Yuan Huang, Yang-Chi-Dung Lin, Jing Li, Kai-Yao Huang, Sirjana Shrestha, Hsiao-Chin Hong, Yun Tang, Yi-Gang Chen, Chen-Nan Jin, Yuan Yu, et al. mirtarbase 2020: updates to the experimentally validated microrna–target interaction database. *Nucleic acids research*, 48(D1):D148–D154, 2020.

[106] Rituparno Sen, Suman Ghosal, Shaoli Das, Subrata Balti, and Jayprokas Chakrabarti. Competing endogenous RNA: the key to posttranscriptional regulation. *The Scientific World Journal*, 2014, 2014.

[107] Denny Zhou, Olivier Bousquet, Thomas N Lal, Jason Weston, and Bernhard Schölkopf. Learning with local and global consistency. In *Advances in neural information processing systems*, pages 321–328, 2004.

[108] TaeHyun Hwang, Hugues Sicotte, Ze Tian, Baolin Wu, Jean-Pierre Kocher, Dennis A Wigle, Vipin Kumar, and Rui Kuang. Robust and efficient identification of biomarkers by classifying features on graphs. *Bioinformatics*, 24(18):2023–2029, 2008.

[109] Wei Zhang, Nicholas Johnson, Baolin Wu, and Rui Kuang. Signed network propagation for detecting differential gene expressions and DNA copy number variations. In *Proceedings of the ACM Conference on Bioinformatics, Computational Biology and Biomedicine*, pages 337–344. ACM, 2012.

[110] Karen CB Souza, Adriane F Evangelista, Letícia F Leal, Cristiano P Souza, René A Vieira, Rhafaela L Causin, AC Neuber, Daniele P Pessoa, Geraldo AS Passos, Rui Reis, et al. Identification of Cell-Free Circulating MicroRNAs for the Detection of Early Breast Cancer and Molecular Subtyping. *Journal of Oncology*, 2019, 2019.

[111] Huanhuan Zhao, Ang Gao, Zhiqian Zhang, Ruoyu Tian, Ang Luo, Mei Li, Dan Zhao, Liya Fu, Li Fu, Jin-Tang Dong, et al. Genetic analysis and preliminary function study of miR-423 in breast cancer. *Tumor Biology*, 36(6):4763–4771, 2015.

[112] Li Ma. Role of miR-10b in breast cancer metastasis. *Breast cancer research*, 12(5):210, 2010.

[113] Himanshu Arora, Rehana Qureshi, and Woong-Yang Park. miR-506 regulates epithelial mesenchymal transition in breast cancer cell lines. *PloS one*, 8(5):e64273, 2013.

[114] Xiaoxiang Chen, Kaixuan Zeng, Mu Xu, Xiangxiang Liu, Xiuxiu Hu, Tao Xu, Bangshun He, Yuqin Pan, Huiling Sun, and Shukui Wang. P53-induced miR-1249 inhibits tumor growth, metastasis, and angiogenesis by targeting VEGFA and HMGA2. *Cell death & disease*, 10(2):1–15, 2019.

[115] Wen Luo, Yuanlong Lin, Shanshan Meng, Yuening Guo, Jiawen Zhang, and Wei Zhang. miRNA-296-3p modulates chemosensitivity of lung cancer cells by targeting CX3CR1. *American journal of translational research*, 8(4):1848, 2016.

[116] Wei Wang, Yan Dong, Xiaoyan Li, Yingying Pan, Jiexin Du, and Daotong Liu. MicroRNA-431 serves as a tumor inhibitor in breast cancer through targeting FGF9. *Oncology Letters*, 19(1):1001–1007, 2020.

[117] Jie Li, Wen Peng, Peng Yang, Ranran Chen, Qiou Gu, Wenwei Qian, Dongjian Ji, Qingyuan Wang, Zhiyuan Zhang, Junwei Tang, et al. MicroRNA-1224-5p inhibits metastasis and epithelial-mesenchymal transition in colorectal cancer by targeting SP1-mediated NF-κB signaling pathways. *Frontiers in Oncology*, 10:294, 2020.

[118] Neha Nagpal, Hafiz M Ahmad, Shibu Chameettachal, Durai Sundar, Sourabh Ghosh, and Ritu Kulshreshtha. HIF-inducible miR-191 promotes migration in breast cancer through complex regulation of TGFβ-signaling in hypoxic microenvironment. *Scientific reports*, 5(1):1–14, 2015.

[119] Ning An, Xinmei Luo, Ming Zhang, and Ruilian Yu. MicroRNA-376b promotes breast cancer metastasis by targeting Hoxd10 directly. *Experimental and therapeutic medicine*, 13(1):79–84, 2017.

[120] Wei-Ting Kuo, Shou-Yu Yu, Sung-Chou Li, Hing-Chung Lam, Hong-Tai Chang, Wei-Shone Chen, Chung-Yu Yeh, Syue-Fen Hung, Tsai-Chi Liu, Tony Wu, et al. MicroRNA-324 in human cancer: miR-324-5p and miR-324-3p have distinct biological functions in human cancer. *Anticancer research*, 36(10):5189–5196, 2016.

[121] Shihua Wang, Chunjing Bian, Zhuo Yang, Ye Bo, Jing Li, Lifen Zeng, Hong Zhou, and Robert Chunhua Zhao. miR-145 inhibits breast cancer cell growth through RTKN. *International journal of oncology*, 34(5):1461–1466, 2009.

[122] Jingwen Chen, Miao Wang, Mingzhou Guo, Yuntao Xie, and Yu-Sheng Cong. miR-127 regulates cell proliferation and senescence by targeting BCL6. *PloS one*, 8(11), 2013.

[123] Hui Xu, Dan Fei, Shan Zong, and Zhimin Fan. MicroRNA-154 inhibits growth and invasion of breast cancer cells through targeting E2F5. *American journal of translational research*, 8(6):2620, 2016.

[124] Xi Gu, Jin-Qi Xue, Si-Jia Han, Song-Ying Qian, and Wen-Hai Zhang. Circulating microRNA-451 as a predictor of resistance to neoadjuvant chemotherapy in breast cancer. *Cancer Biomarkers*, 16(3):395–403, 2016.

[125] Feng Yuan and Wei Wang. MicroRNA-802 suppresses breast cancer proliferation through downregulation of FoxM1. *Molecular medicine reports*, 12(3):4647–4651, 2015.

[126] Y Lu, T Qin, J ea Li, L Wang, Q Zhang, Z Jiang, and J Mao. MicroRNA-140-5p inhibits invasion and angiogenesis through targeting VEGF-A in breast cancer. *Cancer gene therapy*, 24(9):386–392, 2017.

[127] Li-Xu Yan, Xiu-Fang Huang, Qiong Shao, MA-Yan Huang, Ling Deng, Qiu-Liang Wu, Yi-Xin Zeng, and Jian-Yong Shao. MicroRNA miR-21 overexpression in human breast

cancer is associated with advanced clinical stage, lymph node metastasis and patient poor prognosis. *Rna*, 14(11):2348–2360, 2008.

[128] Chen Wang, Zhen Bian, Da Wei, and Jian-guo Zhang. miR-29b regulates migration of human breast cancer cells. *Molecular and cellular biochemistry*, 352(1-2):197–207, 2011.

[129] W Kong, L He, EJ Richards, S Challa, CX Xu, J Permuth-Wey, JM Lancaster, D Coppola, TA Sellers, JY Djeu, et al. Upregulation of miRNA-155 promotes tumour angiogenesis by targeting VHL and is associated with poor prognosis and triple-negative breast cancer. *Oncogene*, 33(6):679–689, 2014.

[130] Hongjiang Wang, Guang Tan, Lei Dong, Lei Cheng, Kejun Li, Zhongyu Wang, and Haifeng Luo. Circulating MiR-125b as a marker predicting chemoresistance in breast cancer. *PloS one*, 7(4), 2012.

[131] Tyler E Miller, Kalpana Ghoshal, Bhuvaneswari Ramaswamy, Satavisha Roy, Jharna Datta, Charles L Shapiro, Samson Jacob, and Sarmila Majumder. MicroRNA-221/222 confers tamoxifen resistance in breast cancer by targeting p27Kip1. *Journal of biological chemistry*, 283(44):29897–29903, 2008.

[132] Enders KO Ng, Rufina Li, Vivian Y Shin, Jennifer M Siu, Edmond SK Ma, and Ava Kwong. MicroRNA-143 is downregulated in breast cancer and regulates DNA methyltransferases 3A in breast cancer cells. *Tumor Biology*, 35(3):2591–2598, 2014.

[133] Yong Li, Maoxiang Zhang, Huijun Chen, Zheng Dong, Vadivel Ganapathy, Muthusamy Thangaraju, and Shuang Huang. Ratio of miR-196s to HOXC8 messenger RNA correlates with breast cancer cell migration and metastasis. *Cancer research*, 70(20):7894–7904, 2010.

[134] Yue Yu, Wei Luo, Zheng-Jun Yang, Jiang-Rui Chi, Yun-Rui Li, Yu Ding, Jie Ge, Xin Wang, and Xu-Chen Cao. miR-190 suppresses breast cancer metastasis by regulation of TGF-β-induced epithelial–mesenchymal transition. *Molecular cancer*, 17(1):70, 2018.

[135] Douglas R Hurst, Mick D Edmonds, Gary K Scott, Christopher C Benz, Kedar S Vaidya, and Danny R Welch. Breast cancer metastasis suppressor 1 up-regulates miR-146, which suppresses breast cancer metastasis. *Cancer research*, 69(4):1279–1283, 2009.

[136] CH Gattolliat, L Thomas, SA Ciafre, G Meurice, G Le Teuff, B Job, C Richon, V Combaret, P Dessen, D Valteau-Couanet, et al. Expression of miR-487b and miR-410 encoded by 14q32. 31 locus is a prognostic marker in neuroblastoma. *British journal of cancer*, 105(9):1352–1361, 2011.

[137] Robert A Smith, Dominik J Jedlinski, Plamena N Gabrovska, Stephen R Weinstein, Larisa Haupt, and Lyn R Griffiths. A genetic variant located in miR-423 is associated with reduced breast cancer risk. *Cancer Genomics-Proteomics*, 9(3):115–118, 2012.

[138] Chong-Zhen Qin, Xiao-Ya Lou, Qiao-Li Lv, Lin Cheng, Na-Yiyuan Wu, Lei Hu, and Hong-Hao Zhou. MicroRNA-184 acts as a potential diagnostic and prognostic marker in epithelial ovarian cancer and regulates cell proliferation, apoptosis and inflammation. *Die Pharmazie-An International Journal of Pharmaceutical Sciences*, 70(10):668–673, 2015.

[139] HS Xu, HL Zong, M Shang, X Ming, JP Zhao, C Ma, and L Cao. MiR-324-5p inhibits proliferation of glioma by target regulation of GLI1. *Eur Rev Med Pharmacol Sci*, 18(6):828–832, 2014.

[140] Ikue Nakayama, Masahiko Shibazaki, Akiko Yashima-Abo, Fumiharu Miura, Toru Sugiyama, Tomoyuki Masuda, and Chihaya Maesawa. Loss of HOXD10 expression induced by upregulation of miR-10b accelerates the migration and invasion activities of ovarian cancer cells. *International journal of oncology*, 43(1):63–71, 2013.

[141] Bingxiang Xiao, Li Tan, Benfu He, Zhiliang Liu, and Ruxiang Xu. MiRNA-329 targeting E2F1 inhibits cell proliferation in glioma cells. *Journal of translational medicine*, 11(1):172, 2013.

[142] H Kang, C Kim, H Lee, JG Rho, JW Seo, Jin-Wu Nam, WK Song, SW Nam, W Kim, and EK Lee. Downregulation of microRNA-362-3p and microRNA-329 promotes tumor progression in human breast cancer. *Cell Death & Differentiation*, 23(3):484–495, 2016.

[143] Bo Sun, Juan Hua, Hongwei Cui, Hongfeng Liu, Kang Zhang, and Haiyan Zhou. MicroRNA-1197 downregulation inhibits proliferation and migration in human non-small cell lung cancer cells by upregulating HOXC11. *Biomedicine & Pharmacotherapy*, 117:109041, 2019.

[144] Yu-Ming Yeh, Chi-Mu Chuang, Kuan-Chong Chao, and Lu-Hai Wang. MicroRNA-138 suppresses ovarian cancer cell invasion and metastasis by targeting SOX4 and HIF-1α. *International journal of cancer*, 133(4):867–878, 2013.

[145] Fan Wang, Jeremy T-H Chang, Chester Jingshiu Kao, and R Stephanie Huang. High expression of miR-532-5p, a tumor suppressor, leads to better prognosis in ovarian cancer both in vivo and in vitro. *Molecular cancer therapeutics*, 15(5):1123–1131, 2016.

[146] Hong Tan, Qingnan He, Guanhui Gong, Yixuan Wang, Juanni Li, Junpu Wang, Ding Zhu, and Xiaoying Wu. miR-382 inhibits migration and invasion by targeting ROR1 through regulating EMT in ovarian cancer. *International journal of oncology*, 48(1):181–190, 2016.

[147] Yong-Wan Kim, Eun Young Kim, Doin Jeon, Juinn-Lin Liu, Helena Suhyun Kim, Jin Woo Choi, and Woong Shick Ahn. Differential microRNA expression signatures and cell type-specific association with Taxol resistance in ovarian cancer cells. *Drug design, development and therapy*, 8:293, 2014.

[148] Xiaolan Zhu, Yuefeng Li, Chanjuan Xie, Xinming Yin, Yueqin Liu, Yuan Cao, Yue Fang, Xin Lin, Yao Xu, Wenlin Xu, et al. miR-145 sensitizes ovarian cancer cells to paclitaxel by targeting Sp1 and Cdk6. *International journal of cancer*, 135(6):1286–1296, 2014.

[149] John K Chan, Kevin Blansit, Tuyen Kiet, Alexander Sherman, Gabriel Wong, Christine Earle, and Lilly YW Bourguignon. The inhibition of miR-21 promotes apoptosis and chemosensitivity in ovarian cancer. *Gynecologic oncology*, 132(3):739–744, 2014.

[150] Qihui Wu, Xiaolei Ren, Yimin Zhang, Xiaodan Fu, Yimin Li, Yulong Peng, Qing Xiao, Tong Li, Chunli Ouyang, Yixi Hu, et al. MiR-221-3p targets ARF4 and inhibits the proliferation and migration of epithelial ovarian cancer cells. *Biochemical and biophysical research communications*, 497(4):1162–1170, 2018.

[151] Richard Flavin, Paul Smyth, Ciara Barrett, S Russell, Hannah Wen, Jianjun Wei, Alex Laios, Sharon O'Toole, M Ring, K Denning, et al. miR-29b expression is associated with disease-free survival in patients with ovarian serous carcinoma. *International Journal of Gynecologic Cancer*, 19(4), 2009.

[152] Silvia Prislei, Enrica Martinelli, Marisa Mariani, Giuseppina Raspaglio, Steven Sieber, Gabriella Ferrandina, Shohreh Shahabi, Giovanni Scambia, and Cristiano Ferlini. MiR-200c and HuR in ovarian cancer. *BMC cancer*, 13(1):72, 2013.

[153] Xiaoying Tian, Limian Xu, and Peng Wang. MiR-191 inhibits TNF-α induced apoptosis of ovarian endometriosis and endometrioid carcinoma cells by targeting DAPK1. *International journal of clinical and experimental pathology*, 8(5):4933, 2015.

[154] Xin Zhou, Fang Zhao, Zhen-Ning Wang, Yong-Xi Song, Hua Chang, Yeunpo Chiang, and Hui-Mian Xu. Altered expression of miR-152 and miR-148a in ovarian cancer is related to cell proliferation. *Oncology reports*, 27(2):447–454, 2012.

[155] Yang Shao, Xiaomin Liu, Jiao Meng, Xiaofei Zhang, Zhongliang Ma, and Gong Yang. MicroRNA-1251-5p promotes carcinogenesis and autophagy via targeting the tumor suppressor TBCC in ovarian cancer cells. *Molecular Therapy*, 27(9):1653–1664, 2019.

[156] Amit K Srivastava, Ananya Banerjee, Tiantian Cui, Chunhua Han, Shurui Cai, Lu Liu, Dayong Wu, Ri Cui, Zaibo Li, Xiaoli Zhang, et al. Inhibition of miR-328–3p Impairs Cancer Stem Cell Function and Prevents Metastasis in Ovarian Cancer. *Cancer research*, 79(9):2314–2326, 2019.

[157] Naima Ahmed Fahmi, Jae-Woong Chang, Heba Nassereddeen, Khandakar Tanvir Ahmed, Deliang Fan, Jeongsik Yong, and Wei Zhang. APA-Scan: Detection and Visualization of 3'-UTR APA with RNA-seq and 3'-end-seq Data. *bioRxiv*, 2020.

[158] Adam J Oates, Lisa M Schumaker, Sara B Jenkins, Amelia A Pearce, Stacey A DaCosta, Banu Arun, and Matthew JC Ellis. The mannose 6-phosphate/insulin-like growth factor 2 receptor (M6P/IGF2R), a putative breast tumor suppressor gene. *Breast cancer research and treatment*, 47(3):269–281, 1998.

[159] SukYeong Jeong, SunYoung Lim, Galina Schevzov, Peter W Gunning, and David M Helfman. Loss of Tpm4. 1 leads to disruption of cell-cell adhesions and invasive behavior in breast epithelial cells via increased Rac1 signaling. *Oncotarget*, 8(20):33544, 2017.

[160] Gareth Watkins, Anthony Douglas-Jones, Richard Bryce, Robert E Mansel, and Wen G Jiang. Increased levels of SPARC (osteonectin) in human breast cancer tissues and its association with clinical outcomes. *Prostaglandins, leukotrienes and essential fatty acids*, 72(4):267–272, 2005.

[161] Yuhong Sun, Xuefei Bao, Yong Ren, Lina Jia, Shenglan Zou, Jian Han, Mengyue Zhao, Mei Han, Hong Li, Qixiang Hua, et al. Targeting HDAC/OAZ1 axis with a novel inhibitor

effectively reverses cisplatin resistance in non-small cell lung cancer. *Cell death & disease*, 10(6):1–13, 2019.

[162] Subbaya Subramanian. Competing endogenous RNAs (ceRNAs): new entrants to the intricacies of gene regulation. *Frontiers in genetics*, 5:8, 2014.

[163] Yuwei Liu, Mengzhu Xue, Shaowei Du, Wanwan Feng, Ke Zhang, Liwen Zhang, Haiyue Liu, Guoyi Jia, Lingshuang Wu, Xin Hu, et al. Competitive endogenous RNA is an intrinsic component of EMT regulatory circuits and modulates EMT. *Nature communications*, 10(1):1–12, 2019.

[164] Zheng Xia, Lawrence A Donehower, Thomas A Cooper, Joel R Neilson, David A Wheeler, Eric J Wagner, and Wei Li. Dynamic Analyses of Alternative Polyadenylation from RNA-Seq Reveal 3'-UTR Landscape Across 7 Tumor Types. *Nature communications*, 5:5274, 2014.

[165] Jae-Woong Chang, Wei Zhang, Hsin-Sung Yeh, Meeyeon Park, Chengguo Yao, Yongsheng Shi, Rui Kuang, and Jeongsik Yong. An integrative model for alternative polyadenylation, IntMAP, delineates mTOR-modulated endoplasmic reticulum stress response. *Nucleic acids research*, 46(12):5996–6008, 2018.

[166] Congting Ye, Yuqi Long, Guoli Ji, Qingshun Quinn Li, and Xiaohui Wu. APAtrap: identification and quantification of alternative polyadenylation sites from RNA-seq data. *Bioinformatics*, 34(11):1841–1849, 2018.

[167] Khandakar Tanvir Ahmed, Sudipto Baul, Yanjie Fu, and Wei Zhang. Attention-based multimodal missing value imputation for time series data with high missing rate. In *Proceedings of the 2023 SIAM International Conference on Data Mining (SDM)*, pages 469–477. SIAM, 2023.

[168] Louis-Pascal Xhonneux, Oliver Knight, Åke Lernmark, Ezio Bonifacio, William A Hagopian, Marian J Rewers, Jin-Xiong She, Jorma Toppari, Hemang Parikh, Kenneth GC Smith, et al. Transcriptional networks in at-risk individuals identify signatures of type 1 diabetes progression. *Science translational medicine*, 13(587):eabd5666, 2021.

[169] Bobbie-Jo M Webb-Robertson, Lisa M Bramer, Bryan A Stanfill, Sarah M Reehl, Ernesto S Nakayasu, Thomas O Metz, Brigitte I Frohnert, Jill M Norris, Randi K Johnson, Stephen S Rich, et al. Prediction of the development of islet autoantibodies through integration of environmental, genetic, and metabolic markers. *Journal of diabetes*, 13(2):143–153, 2021.

[170] Sidra Mehtab and Jaydip Sen. Analysis and forecasting of financial time series using CNN and LSTM-based deep learning models. In *Advances in Distributed Computing and Machine Learning*, pages 405–423. Springer, 2022.

[171] Bryan Lim and Stefan Zohren. Time-series forecasting with deep learning: a survey. *Philosophical Transactions of the Royal Society A*, 379(2194):20200209, 2021.

[172] Yuxiu Hua, Zhifeng Zhao, Rongpeng Li, Xianfu Chen, Zhiming Liu, and Honggang Zhang. Deep learning with long short-term memory for time series prediction. *IEEE Communications Magazine*, 57(6):114–119, 2019.

[173] Yonghong Luo, Xiangrui Cai, Ying Zhang, Jun Xu, et al. Multivariate time series imputation with generative adversarial networks. *Advances in neural information processing systems*, 31, 2018.

[174] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[175] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

[176] Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Comput.*, 9(8):1735–1780, nov 1997.

[177] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

[178] Eunkyu Oh, Taehun Kim, Yunhu Ji, and Sushil Khyalia. STING: Self-attention based Time-series Imputation Networks using GAN. In *2021 IEEE International Conference on Data Mining (ICDM)*, pages 1264–1269. IEEE, 2021.

[179] Wenjie Du, David Cote, and Yan Liu. SAITS: Self-Attention-based Imputation for Time Series. 2022.

[180] Elizabeth Brunk, Kevin W George, Jorge Alonso-Gutierrez, Mitchell Thompson, Edward Baidoo, George Wang, Christopher J Petzold, Douglas McCloskey, Jonathan Monk, Laurence Yang, et al. Characterizing strain variation in engineered E. coli using a multi-omics-based workflow. *Cell systems*, 2(5):335–346, 2016.

[181] AJ Newman, MP Clark, Kevin Sampson, Andrew Wood, LE Hay, A Bock, RJ Viger, D Blodgett, L Brekke, JR Arnold, et al. Development of a large-sample watershed-scale hydrometeorological data set for the contiguous USA: data set characteristics and assessment of regional variability in hydrologic model performance. *Hydrology and Earth System Sciences*, 19(1):209–223, 2015.

[182] TEDDY Study Group. The Environmental Determinants of Diabetes in the Young (TEDDY) study: study design. *Pediatric diabetes*, 8(5):286–298, 2007.

[183] N. Addor, A. J. Newman, N. Mizukami, and M. P. Clark. The CAMELS data set: catchment attributes and meteorology for large-sample studies. *Hydrology and Earth System Sciences*, 21(10):5293–5313, 2017.

[184] Andrew J Newman, Naoki Mizukami, Martyn P Clark, Andrew W Wood, Bart Nijssen, and Grey Nearing. Benchmarking of a physically based hydrologic model. *Journal of Hydrometeorology*, 18(8):2215–2225, 2017.

[185] Wei Cao, Dong Wang, Jian Li, Hao Zhou, Lei Li, and Yitan Li. Brits: Bidirectional recurrent imputation for time series. *Advances in neural information processing systems*, 31, 2018.

[186] T Konstantin Rusch and Siddhartha Mishra. UnICORNN: A recurrent model for learning very long time dependencies. In *International Conference on Machine Learning*, pages 9168–9178. PMLR, 2021.

[187] Rui Xie, Jia Wen, Andrew Quitadamo, Jianlin Cheng, and Xinghua Shi. A deep auto-encoder model for gene expression prediction. *BMC genomics*, 18(9):39–49, 2017.

[188] Khandakar Tanvir Ahmed, Sze Cheng, Qian Li, Jeongsik Yong, and Wei Zhang. Incomplete time-series gene expression in integrative study for islet autoimmunity prediction. *Briefings in Bioinformatics*, 24(1):bbac537, 2023.

[189] Benjamin F Crabtree, Subhash C Ray, Priscilla M Schmidt, Patrick T O'Connor, and David D Schmidt. The individual over time: time series applications in health care research. *Journal of clinical epidemiology*, 43(3):241–260, 1990.

[190] Anne M Euser, Carmine Zoccali, Kitty J Jager, and Friedo W Dekker. Cohort studies: prospective versus retrospective. *Nephron Clinical Practice*, 113(3):c214–c217, 2009.

[191] Samer Hammoudeh, Wessam Gadelhaq, and Ibrahim Janahi. *Prospective cohort studies in medical research*. IntechOpen, 2018.

[192] Muhammad Saad, Mohita Chaudhary, Fakhri Karray, and Vincent Gaudet. Machine learning based approaches for imputation in time series data and their impact on forecasting. In *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 2621–2627. IEEE, 2020.

[193] Md Bahadur Badsha, Rui Li, Boxiang Liu, Yang I Li, Min Xian, Nicholas E Banovich, and Audrey Qiuyan Fu. Imputation of single-cell gene expression with an autoencoder neural network. *Quantitative Biology*, 8(1):78–94, 2020.

[194] Ramon Viñas, Tiago Azevedo, Eric R Gamazon, and Pietro Liò. Deep Learning Enables Fast and Accurate Imputation of Gene Expression. *Frontiers in genetics*, 12:489, 2021.

[195] Kohbalan Moorthy, Mohd Saberi Mohamad, and Safaai Deris. A review on missing value imputation algorithms for microarray gene expression data. *Current Bioinformatics*, 9(1):18–22, 2014.

[196] Valentin Voillet, Philippe Besse, Laurence Liaubet, Magali San Cristobal, and Ignacio González. Handling missing rows in multi-omics data integration: multiple imputation in multiple factor analysis framework. *BMC bioinformatics*, 17(1):1–16, 2016.

[197] Dongdong Lin, Jigang Zhang, Jingyao Li, Chao Xu, Hong-Wen Deng, and Yu-Ping Wang. An integrative imputation method based on multi-omics datasets. *BMC bioinformatics*, 17(1):1–12, 2016.

[198] Kohbalan Moorthy, Aws Naser Jaber, Mohd Arfian Ismail, Ferda Ernawan, Mohd Saberi Mohamad, and Safaai Deris. Missing-values imputation algorithms for microarray gene expression data. *Microarray Bioinformatics*, pages 255–266, 2019.

[199] Miew Keen Choong, Maurice Charbit, and Hong Yan. Autoregressive-model-based missing value estimation for DNA microarray time series data. *IEEE Transactions on information technology in biomedicine*, 13(1):131–137, 2009.

[200] Eben Afrifa-Yamoah, Ute A Mueller, SM Taylor, and AJ Fisher. Missing data imputation of high-resolution temporal climate time series data. *Meteorological Applications*, 27(1):e1873, 2020.

[201] Teddy. *The environmental determinants of diabetes in the young (TEDDY) study*, n.d. `https://teddy.epi.usf.edu/` [Accessed: 2021-12-31].

[202] Eiji Kawasaki. Type 1 diabetes and autoimmunity. *Clinical pediatric endocrinology*, 23(4):99–105, 2014.

[203] Jeffrey P Krischer, Xiang Liu, Kendra Vehik, Beena Akolkar, William A Hagopian, Marian J Rewers, Jin-Xiong She, Jorma Toppari, Anette-G Ziegler, Åke Lernmark, et al. Predicting islet cell autoimmunity and type 1 diabetes: an 8-year TEDDY study progress report. *Diabetes care*, 42(6):1051–1060, 2019.

[204] Matej Orešič, Peddinti Gopalacharyulu, Juha Mykkänen, Niina Lietzen, Marjaana Mäkinen, Heli Nygren, Satu Simell, Ville Simell, Heikki Hyöty, Riitta Veijola, et al. Cord serum lipidome in prediction of islet autoimmunity and type 1 diabetes. *Diabetes*, 62(9):3268–3274, 2013.

[205] Christiane Winkler, Jan Krumsiek, Florian Buettner, Christof Angermüller, Eleni Z Giannopoulou, Fabian J Theis, Anette-Gabriele Ziegler, and Ezio Bonifacio. Feature ranking of type 1 diabetes susceptibility genes improves prediction of type 1 diabetes. *Diabetologia*, 57(12):2521–2529, 2014.

[206] Richard A Oram, Kashyap Patel, Anita Hill, Beverley Shields, Timothy J McDonald, Angus Jones, Andrew T Hattersley, and Michael N Weedon. A type 1 diabetes genetic risk score can aid discrimination between type 1 and type 2 diabetes in young adults. *Diabetes care*, 39(3):337–344, 2016.

[207] Andreas Beyerlein, Ezio Bonifacio, Kendra Vehik, Markus Hippich, Christiane Winkler, Brigitte I Frohnert, Andrea K Steck, William A Hagopian, Jeffrey P Krischer, Åke Lernmark, et al. Progression from islet autoimmunity to clinical type 1 diabetes is influenced by genetic factors: results from the prospective TEDDY study. *Journal of medical genetics*, 56(9):602–605, 2019.

[208] Ezio Bonifacio, Andreas Beyerlein, Markus Hippich, Christiane Winkler, Kendra Vehik, Michael N Weedon, Michael Laimighofer, Andrew T Hattersley, Jan Krumsiek, Brigitte I Frohnert, et al. Genetic scores to stratify risk of developing multiple islet autoantibodies and type 1 diabetes: a prospective study in children. *PLoS medicine*, 15(4):e1002548, 2018.

[209] Markus Hippich, Andreas Beyerlein, William A Hagopian, Jeffrey P Krischer, Kendra Vehik, Jan Knoop, Christiane Winker, Jorma Toppari, Åke Lernmark, Marian J Rewers, et al. Genetic contribution to the divergence in type 1 diabetes risk between children from the general population and children from affected families. *Diabetes*, 68(4):847–857, 2019.

[210] Jay M Sosenko, Jerry P Palmer, Lisa Rafkin-Mervis, Jeffrey P Krischer, David Cuthbertson, Della Matheson, and Jay S Skyler. Glucose and C-peptide changes in the perionset period of type 1 diabetes in the Diabetes Prevention Trial–Type 1. *Diabetes Care*, 31(11):2188–2192, 2008.

[211] Maria J Redondo, Susan Geyer, Andrea K Steck, Seth Sharp, John M Wentworth, Michael N Weedon, Peter Antinozzi, Jay Sosenko, Mark Atkinson, Alberto Pugliese, et al. A type 1

diabetes genetic risk score predicts progression of islet autoimmunity and development of type 1 diabetes in individuals at risk. *Diabetes care*, 41(9):1887–1894, 2018.

[212] Lauric A Ferrat, Kendra Vehik, Seth A Sharp, Åke Lernmark, Marian J Rewers, Jin-Xiong She, Anette-G Ziegler, Jorma Toppari, Beena Akolkar, Jeffrey P Krischer, et al. A combined risk score enhances prediction of type 1 diabetes among susceptible children. *Nature medicine*, 26(8):1247–1255, 2020.

[213] Michael D Radmacher, Lisa M McShane, and Richard Simon. A paradigm for class prediction using gene expression profiles. *Methods*, 2018.

[214] Ran Su, Xinyi Liu, Leyi Wei, and Quan Zou. Deep-Resp-Forest: A deep forest model to predict anti-cancer drug response. *Methods*, 166:91–102, 2019.

[215] Kourosh Zarringhalam, David Degras, Christoph Brockel, and Daniel Ziemek. Robust phenotype prediction from gene expression data using differential shrinkage of co-regulated genes. *Scientific reports*, 8(1):1–10, 2018.

[216] Christos Dimitrakopoulos, Sravanth Kumar Hindupur, Luca Häfliger, Jonas Behr, Hesam Montazeri, Michael N Hall, and Niko Beerenwinkel. Network-based integration of multi-omics data for prioritizing cancer genes. *Bioinformatics*, 34(14):2441–2448, 2018.

[217] Amrit Singh, Casey P Shannon, Benoît Gautier, Florian Rohart, Michaël Vacher, Scott J Tebbutt, and Kim-Anh Lê Cao. DIABLO: an integrative approach for identifying key molecular drivers from multi-omics assays. *Bioinformatics*, 35(17):3055–3062, 2019.

[218] Kumardeep Chaudhary, Olivier B Poirion, Liangqun Lu, and Lana X Garmire. Deep learning–based multi-omics integration robustly predicts survival in liver cancer. *Clinical Cancer Research*, 24(6):1248–1259, 2018.

[219] Khandakar Tanvir Ahmed, Jiao Sun, Sze Cheng, Jeongsik Yong, and Wei Zhang. Multi-omics data integration by generative adversarial network. *Bioinformatics*, 38(1):179–186, 08 2021.

[220] Laura M Jacobsen, Helena E Larsson, Roy N Tamura, Kendra Vehik, Joanna Clasen, Jay Sosenko, William A Hagopian, Jin-Xiong She, Andrea K Steck, Marian Rewers, et al. Predicting progression to type 1 diabetes from ages 3 to 6 in islet autoantibody positive TEDDY children. *Pediatric diabetes*, 20(3):263–270, 2019.

[221] Qian Li, Hemang Parikh, Martha D Butterworth, Åke Lernmark, William Hagopian, Marian Rewers, Jin-Xiong She, Jorma Toppari, Anette-G Ziegler, Beena Akolkar, et al. Longitudinal metabolome-wide signals prior to the appearance of a first islet autoantibody in children participating in the TEDDY study. *Diabetes*, 69(3):465–476, 2020.

[222] Andrea K Steck, Kendra Vehik, Ezio Bonifacio, Ake Lernmark, Anette-G Ziegler, William A Hagopian, JinXiong She, Olli Simell, Beena Akolkar, Jeffrey Krischer, et al. Predictors of progression from the appearance of islet autoantibodies to early childhood diabetes: The Environmental Determinants of Diabetes in the Young (TEDDY). *Diabetes care*, 38(5):808–813, 2015.

[223] Jie Cai, Jiawei Luo, Shulin Wang, and Sheng Yang. Feature selection in machine learning: A new perspective. *Neurocomputing*, 300:70–79, 2018.

[224] Bissan Ghaddar and Joe Naoum-Sawaya. High dimensional data classification and feature selection using support vector machines. *European Journal of Operational Research*, 265(3):993–1004, 2018.

[225] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher,

M. Perrot, and E. Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[226] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32:8026–8037, 2019.

[227] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[228] Dean G Brown, Heike J Wobst, Abhijeet Kapoor, Leslie A Kenna, and Noel Southall. Clinical development times for innovative drugs. *Nat. Rev. Drug Discov*, 21(11):793–794, 2021.

[229] Olivier J Wouters, Martin McKee, and Jeroen Luyten. Estimated research and development investment needed to bring a new medicine to market, 2009-2018. *Jama*, 323(9):844–853, 2020.

[230] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[231] Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

[232] Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, Ghalia Rehawi, Yu Wang, Llion Jones, Tom Gibbs, Tamas Feher, Christoph Angerer, Martin Steinegger, et al. Prottrans: To-

ward understanding the language of life through self-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 44(10):7112–7127, 2021.

[233] Nadav Brandes, Dan Ofer, Yam Peleg, Nadav Rappoport, and Michal Linial. ProteinBERT: a universal deep-learning model of protein sequence and function. *Bioinformatics*, 38(8):2102–2110, 2022.

[234] Seyone Chithrananda, Gabriel Grand, and Bharath Ramsundar. ChemBERTa: large-scale self-supervised pretraining for molecular property prediction. *arXiv preprint arXiv:2010.09885*, 2020.

[235] Nathan C Frey, Ryan Soklaski, Simon Axelrod, Siddharth Samsi, Rafael Gomez-Bombarelli, Connor W Coley, and Vijay Gadepally. Neural scaling of deep chemical models. *Nature Machine Intelligence*, pages 1–9, 2023.

[236] Jerret Ross, Brian Belgodere, Vijil Chenthamarakshan, Inkit Padhi, Youssef Mroueh, and Payel Das. Large-scale chemical language representations capture molecular structure and properties. *Nature Machine Intelligence*, 4(12):1256–1264, 2022.

[237] Tri Minh Nguyen, Thin Nguyen, and Truyen Tran. Mitigating cold-start problems in drug-target affinity prediction with interaction knowledge transferring. *Briefings in Bioinformatics*, 23(4):bbac269, 2022.

[238] Yogesh Kalakoti, Shashank Yadav, and Durai Sundar. TransDTI: transformer-based language models for estimating DTIs and building a drug recommendation workflow. *ACS omega*, 7(3):2706–2717, 2022.

[239] Hyeunseok Kang, Sungwoo Goo, Hyunjung Lee, Jung-woo Chae, Hwi-yeol Yun, and Sangkeun Jung. Fine-tuning of bert model to accurately predict drug–target interactions. *Pharmaceutics*, 14(8):1710, 2022.

[240] Fangping Wan, Lixiang Hong, An Xiao, Tao Jiang, and Jianyang Zeng. NeoDTI: neural integration of neighbor information from a heterogeneous network for discovering new drug–target interactions. *Bioinformatics*, 35(1):104–111, 2019.

[241] Kang Wang, Jing Hu, and Xiaolong Zhang. Identifying Drug–Target Interactions Through a Combined Graph Attention Mechanism and Self-attention Sequence Embedding Model. In *International Conference on Intelligent Computing*, pages 246–257. Springer, 2023.

[242] Shugang Zhang, Mingjian Jiang, Shuang Wang, Xiaofeng Wang, Zhiqiang Wei, and Zhen Li. SAG-DTA: prediction of drug–target affinity using self-attention graph network. *International Journal of Molecular Sciences*, 22(16):8993, 2021.

[243] Haiyang Wang, Guangyu Zhou, Siqi Liu, Jyun-Yu Jiang, and Wei Wang. Drug-target interaction prediction with graph attention networks. *arXiv preprint arXiv:2107.06099*, 2021.

[244] Lu Jiang, Jiahao Sun, Yue Wang, Qiao Ning, Na Luo, and Minghao Yin. Identifying drug–target interactions via heterogeneous graph attention networks combined with cross-modal similarities. *Briefings in Bioinformatics*, 23(2):bbac016, 2022.

[245] Zhongjian Cheng, Cheng Yan, Fang-Xiang Wu, and Jianxin Wang. Drug-target interaction prediction using multi-head self-attention and graph attention network. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 19(4):2208–2218, 2021.

[246] The UniProt Consortium . UniProt: the Universal Protein Knowledgebase in 2023. *Nucleic Acids Research*, 51(D1):D523–D531, 11 2022.

[247] Baris E Suzek, Yuqi Wang, Hongzhan Huang, Peter B McGarvey, Cathy H Wu, and UniProt Consortium. UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics*, 31(6):926–932, 2015.

[248] Sunghwan Kim, Jie Chen, Tiejun Cheng, Asta Gindulyte, Jia He, Siqian He, Qingliang Li, Benjamin A Shoemaker, Paul A Thiessen, Bo Yu, et al. PubChem 2023 update. *Nucleic acids research*, 51(D1):D1373–D1380, 2023.

[249] HuggingFace. Hugging Face, 2023. `https://huggingface.co/` [Accessed: 2023-10-26].

[250] Sameh K Mohamed, Aayah Nounu, and Vít Nováček. Biological applications of knowledge graph embedding models. *Briefings in bioinformatics*, 22(2):1679–1693, 2021.

[251] Vivian Law, Craig Knox, Yannick Djoumbou, Tim Jewison, An Chi Guo, Yifeng Liu, Adam Maciejewski, David Arndt, Michael Wilson, Vanessa Neveu, et al. DrugBank 4.0: shedding new light on drug metabolism. *Nucleic acids research*, 42(D1):D1091–D1097, 2014.

[252] Tiqing Liu, Yuhmei Lin, Xin Wen, Robert N Jorissen, and Michael K Gilson. BindingDB: a web-accessible database of experimentally determined protein–ligand binding affinities. *Nucleic acids research*, 35(suppl_1):D198–D201, 2007.

[253] Yoshihiro Yamanishi, Michihiro Araki, Alex Gutteridge, Wataru Honda, and Minoru Kanehisa. Prediction of drug–target interaction networks from the integration of chemical and genomic spaces. *Bioinformatics*, 24(13):i232–i240, 2008.

[254] Richard Liaw, Eric Liang, Robert Nishihara, Philipp Moritz, Joseph E Gonzalez, and Ion Stoica. Tune: A Research Platform for Distributed Model Selection and Training. *arXiv preprint arXiv:1807.05118*, 2018.

[255] RDKit. RDKit: Open-source cheminformatics, 2023. `https://www.rdkit.org` [Accessed: 2023-10-26].

[256] Fabian Sievers, Andreas Wilm, David Dineen, Toby J Gibson, Kevin Karplus, Weizhong Li, Rodrigo Lopez, Hamish McWilliam, Michael Remmert, Johannes Söding, et al. Fast, scal-

able generation of high-quality protein multiple sequence alignments using clustal omega. *Molecular systems biology*, 7(1):539, 2011.