ECAI 2024
U. Endriss et al. (Eds.)

© 2024 The Authors.
This article is published online with Open Access by IOS Press and distributed under the terms of the Creative Commons Attribution Non-Commercial License 4.0 (CC BY-NC 4.0).
doi:10.3233/FAIA240669

DrDiff: Drug Response Prediction Through Controllable Diffusion-GE and Graph Attention Network

Seungyeon Choi[a,1], Samgmin Seo[a], Jonghwan Choi[b], Chihyun Park[c,*] and Sanghyun Park[a,]**

[a]Department of Computer Science, Yonsei University, Seoul, Republic of Korea
[b]Division of Software, Hallym University, Chuncheon, Republic of Korea
[c]Department of Computer Science and Engineering, Kangwon National University, Chuncheon, Republic of Korea

Abstract. The accurate prediction of drug responses based on the genomic profile of a patient is essential to progress in the field of precision medicine. The advent of various deep-learning algorithms based on publicly available large-scale omics datasets is the driving force behind research in this field. The characteristics of biological datasets, characterized by high dimensions and low sample sizes, pose challenges of overfitting and limited generalization in prediction models. Additionally, constructing prediction models using biological data such as gene expression is further complicated by the need to account for the complex relationships among genes, which exacerbates the aforementioned challenges. To address these challenges, we propose a drug response prediction framework (DrDiff) that integrates a denoising diffusion probabilistic model (DDPM) based data augmentation module with a graph attention network based drug response prediction module. The proposed model showed a 10% higher AUC than the state-of-the-art models for drug response prediction for the six drugs considered in the study, suggesting the superior generalization performance of DrDiff over other baseline models. Furthermore, we demonstrated the feasibility of generative models, which form one of the modules of the proposed framework, in overcoming the fundamental limitations of omics datasets. Further experiments bear out the feasibility of generative models, which form one of the modules of the proposed framework, in augmenting gene expression data.

1 Introduction

Precision oncology, which aims to provide personalized cancer treatments based on the genetic characteristics of individual tumors, avoids ineffective treatments, emerging as a promising approach for improving patient outcomes and reducing healthcare costs [7]. Accurately predicting drug responses, which indicates how specific drug is effective as a treatment, is greatly becoming important in the current precision oncology. However, variabilities in drug responses among patients, which are attributed to genetic variations, make obtaining accurate predictions challenging [25, 29]. Many studies have been conducted on deep learning (DL) models that learn the genetic information of patients for drug response predictions, with many previous studies revealing that Gene Expression(GE) data, which represents the measurement of the activity of genes, is the most effective type of data for drug response predictions [9, 19].

In field of bioinformatics such as cancer subtype or drug response prediction studies using omics data including GE data, high dimension and low sample size of input data are fundamental challenges. Many features or variables with relatively few input data samples increase the risk of overfitting in training datasets and reduce the generality of prediction models [1, 22]. Additionally, the robustness of such models are affected by outliers in datasets with small sample sizes. Therefore, it is essential to carefully balance the number of features and sample sizes when conducting data analysis, particularly for omics data, which are often characterized by large dimensions and small sample sizes; we aimed to overcome these obstacles in this study.

To effectively address the aforementioned problem, approaches from various perspectives have been proposed: *Data Augmentation, Utilizing Inductive Bias*. First possible solution would be augmenting training data by utilizing generative models, such as variational autoencoders (VAEs) [4] and generative adversarial networks (GANs) [18], which have been actively studied in the field of image processing. Notably, there is a study that proposes improvements in cancer classification performance by augmenting GE data using a GAN and one that proposes a universal tabular generative model that can be applied to data such as GE data [5]. However, these studies do not account for generative models neglecting to consider the relationship between genes when learning, which limits their ability to capture biological mechanisms. Additionally the advancement in the field of generative models has led to the emergence of innovative models such as DDPM, which exceed the capabilities of GAN and VAE [14, 30]. These models present new possibilities for utilization in augmentation tasks, which aim to alleviate data sparsity. Lastly, to overcome the degradation of model predictive power due to limited training data, a strategy is to use the relationships between genes as an inductive bias for the predictive model [3, 12]. This approach can be realized by utilizing graph neural networks that can learn the relationships between biological pathways obtainable from various databases [17]. However, in order to increase the predictive power more efficiently with limited data, there is a need for approaches that not only efficiently integrate multiple biological pathways but also better capture biological mechanisms by training models to identify the importance of relationships between genes in predicting drug response.

To overcome the limitations of existing studies, we propose a

* Corresponding Author. Email: chihyun@kangwon.ac.kr
** Corresponding Author. Email: sanghyun@yonsei.ac.kr
[1] First Author. Email: tmddus1553@yonsei.ac.kr

framework capable of addressing the overfitting problem in training data caused by the high dimensionality and low sample sizes in omics datasets in the context of drug response prediction. The proposed framework DrDiff comprises three main modules: 1) feature selection through biological pathway analysis to solve the issue of high dimensionality, 2) GE data augmentation utilizing diffusion-based generative models (Diffusion-GE) to address the issue of low sample sizes, and 3) drug response prediction using graph attention networks. The first module aims to solve the problem of high dimensionality by extracting the biological pathways most closely related to the target protein of each drug and selecting the genes that comprise these pathways. The second module aims to augment GE data by leveraging a recent generative model (denoising diffusion probabilistic model), in which a graph autoencoder (AE) is employed to capture biological mechanisms. The final module employs a graph attention network, which exploits prior knowledge about biological pathways to maximize the generalization of models from limited training data.

The main contributions of the study are summarized as follows.

- To the best of our knowledge, this is the first study to adapt and enhance the recently developed denoising diffusion probabilistic model (DDPM) for developing a GE data augmentation model that captures biological complexities and addresses the issue of omics (gene expression) data scarcity.
- We present a novel approach for highly accurate drug response prediction. Our approach leverages the capabilities of a graph attention network, which effectively combines information from a wide range of biological pathways while also considering the proximity to target proteins.
- The proposed framework, DrDiff, demonstrated superior performance in drug response prediction on patient datasets not seen during training, surpassing other recent methods. Furthermore, we presented evidence of the generative module, a core component of our framework, consistently generating higher quality data compared to data produced by other generative models.

2 Related Work

Numerous approaches have been used for drug response predictions. Traditionally, machine learning techniques have been utilized to select crucial features for prediction [7, 9, 33]. In recent years, with the advancement of various DL techniques, research on drug response predictions has evolved. One such model, multi-omics late integration (MOLI) [27], utilizes a deep neural network architecture that enables the integration of multiple omics data types at different stages of a network. Supervised feature extraction learning using triplet loss (Super.FELT) [23], on the other hand, employs feature selection to reduce the dimensionality of multi-omics data, followed by a supervised encoder that extracts crucial information from the reduced omics dataset. Then, it classifies the encoded omics datasets based on a neural network for drug response prediction. A novel model, DeepInsight-3D [28], converts structured data into images using convolutional neural networks (CNNs) for predicting patient-specific anticancer drug responses.

3 Preliminaries

Definition 1. **Denoising Diffusion Probabilistic Model** A Denoising Diffusion Probabilistic Model (DDPMs) [14] is designed to learn a Markov Chain that systematically transforms a basic probability distribution, often a random Gaussian distribution, into a distribution that resembles actual data. This generative process is essentially the reverse of the forward diffusion process within DDPM. In this forward process, a fixed Markov chain progressively introduces noise to data while sequentially sampling latent variables $x_0, x_1, ..., x_T$, all having the same dimensionality. Each step in the forward process is a Gaussian translation.

$$q\left(x_t \mid x_{t-1}\right) := \mathcal{N}\left(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t \mathbf{I}\right), \tag{1}$$

where $\beta_1, \beta_2, ..., \beta_T$ are predetermined variance settings rather than parameters that the model learns. Eq. (1) calculates x_T by introducing minor Gaussian noise to the latent variable. Given clean data x_0 , the sampling of x_T is expressed in a closed form by reparameterization trick as follows:

$$q\left(x_t \mid x_0\right) = \mathcal{N}\left(x_t; \sqrt{\bar{a}_t}x_0, \left(1-\bar{a}_t\right)\mathbf{I}\right) \tag{2}$$

$$x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon \tag{3}$$

where $\alpha_t := 1 - \beta_t, \quad$ and $\bar{\alpha}_t := \prod_{s=1}^{t} \alpha_s$. And $\epsilon \sim N(0, \mathbf{I})$ is a random gaussian distribution with the same dimensionality as x_0. The training strategy of DDPM commences with the parameterization $p_\theta\left(x_{t-1} \mid x_t\right)$ of Gaussian transitions in the reverse direction of the forward process, which injects noise. Subsequently, the strategy culminates in training the reverse process's mean to predict the mean of the forward process in order to learn the transition kernel of the reverse process.

$$p_\theta\left(x_{t-1} \mid x_t\right) = \mathcal{N}\left(x_{t-1}; \mu_\theta\left(x_t, t\right), \sigma_t^2 \mathbf{I}\right) \tag{4}$$

$$L_{\text{simple}} := \mathbb{E}_{t,\epsilon,x_0}\left[\left\|\epsilon - \epsilon_\theta\left(\sqrt{\bar{a}_t}x_0 + \sqrt{1-\bar{a}_t}\epsilon, t\right)\right\|^2\right] \tag{5}$$

Eq. (2) provides an efficient method to jump directly to an arbitrary step in the forward noise process, making it possible to randomly sample t during training. To predict $\mu_\theta\left(x_t, t\right)$ efficiently, DDPM adopted a specific parameterization approach based on Eq. (4) and Eq. (5). Hence, a trainable neural network could predict the noise added to x_0. These authors found that using ϵ in prediction produced the best results, especially when it was combined with a reweighted loss function (eq.(5)).

Definition 2. **Gene Expression Graph Construction** A set of biological pathways, denoted as $\mathbb{G} = \{\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_k\}$, is defined on a set of subgraphs $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Subgraph $\mathcal{G}$ denote an undirected graph comprised of a set of nodes $v_i \in \mathcal{V}$ and a set of edges $e_{ij} = (v_i, v_j) \in \mathcal{E}$. Each node represents a gene, and an edge represents a relationship between genes.

4 Method

4.1 Framework Design

In this section, we introduce the proposed DrDiff framework for drug response prediction. Figure 1 illustrates the overall process of the proposed framework.

4.2 Network Analysis for Feature Selection

To identify the biological pathways that most significantly impact drug response prediction results, the proximal pathways that are statistically associated with drug-associated genes need to be identified. We calculated the proximal pathway by measuring the distance from the drug-associated genes to each pathway (Figure 1.A) using the method proposed by Guney et al.[11] The measurement was based on

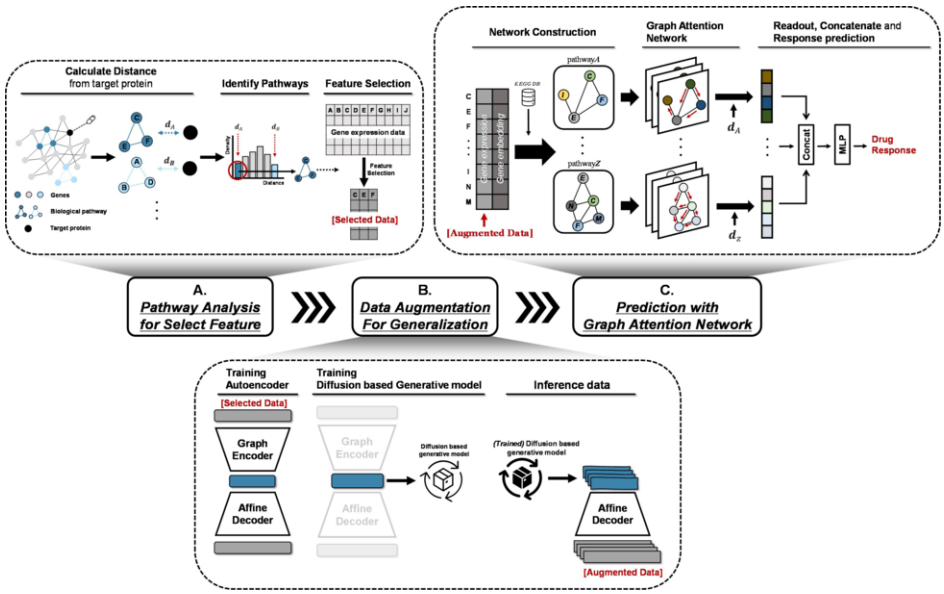


Figure 1. Overview of **DrDiff**.

the average shortest path lengths between the drug-associated genes and the nearest pathway genes, according to the following equation:

$$d_{\text{closet}} = \frac{1}{|T|} \sum_{t \in T} \min_{s \in S} d(s,t) \tag{6}$$

where T represents the set of drug-associated genes (target genes), S represents the pathway genes, and $d(s,t)$ is the shortest path between the drug-associated and pathway genes.

To identify whether the calculated distance for each pathway was statistically significant regardless of the number of nodes (genes) that made up the pathway, random genes were bootstrapped to generate a reference distribution.

We calculated the z-score of the distances for each pathway using the mean and standard deviation of the reference distribution. Subsequently, the pathways with the shortest distances, representing the lowest 10%, were considered to be most closely related to the drug. Finally, among all the genes, those that most significantly impacted drug response predictions, namely those included in the selected pathways, were selected for further analysis.

4.3 Data Augmentation for Generalization

This subsection introduces the second module of the proposed framework, a novel data augmentation method that addresses the lack of generalization due to insufficient training data (Figure 1.B). The module comprises two main components. The first component leverages the graph AE to map gene expression data to the latent space. The second component generates the latent space obtained using the DDPM. Finally, the augmented latent space is converted back to the gene expression data level using the trained decoder of the graph AE (Alg. 1).

4.3.1 Graphical Compression of Gene Expression Profiles The proposed graphical compression model was based on the AE architecture. It took the network information of the biological pathways representing the top K pathways proximal to the target protein extracted through the method described in section 4.2 as an input and encoded this into a graph representation latent space. The gene expression data $X \in \mathbb{R}^{N \times D}$ input to the AE model represented each biological pathway (extracted in Section 4.2) as one subgraph, as shown in section 4.4.1, where the node features of the subgraph comprised gene expression profiles and gene indicators. Likewise, encoder layer is defined with the same architecture as the graph attention layer used in Section 4.4.1, and affine layer was added to produce latent representations $r \in \mathbb{R}^{N \times d}$ $(d \ll D)$ of the input data after the graph attention operation. Subsequently, a decoder comprising affine layers was trained to reconstruct the initial node state, which contained only the gene expression profiles. Training was done by optimizing the L_{GraphAE} loss function with respect to ϕ and ψ as follows:

$$L_{\text{GraphAE}} = \sum_{i=1}^{n} \left(x_i - \texttt{De}_{\phi}\left(\texttt{En}_{\psi}\left(x_i, \mathcal{E}\right)\right)\right)^2 \tag{7}$$

Subsequently, generation model introduced in the next section is designed to generate the latent space extracted in this section.

4.3.2 Generative Modeling of Latent Space . Regarding the trained graph AE model, which comprised $\texttt{En}_{\psi}$ and $\texttt{De}_{\phi}$, it had access to a low-dimensional latent space containing information about biological relationships. Additionally, contrasting with the image generation task in which the DDPM is commonly used, in this study, this model required modifications to generate latent spaces containing gene expression information and the relationships between genes.

In the field of image generation, the backbone architecture of the DDPM-based model is implemented as U-Net because the CNN operations that form the U-Net architecture align well with the inductive bias of image-like data. However, in our case, adopting a CNN-based U-Net model as a backbone architecture is inappropriate because unlike images, the data to be covered do not exhibit locality or dependencies among neighboring pixels. Therefore, considering the characteristics of the gene expression data, the backbone architecture was changed from a convolution layer to an affine layer.

The module discussed in this section enhances the predictive performance of the model in determining the sensitivity or resistance of a sample to a drug by increasing the amount of training data. To achieve this, instead of unconditionally generating samples, samples with labeled indications of their sensitivities or resistances to drugs

Algorithm 1 Overall data augmentation process

Input:
Gene expression data $\leftarrow x$
Response label for the records in $x \leftarrow c$
Edge information, Weight for the AE $\leftarrow \mathcal{E}, \psi, \phi$
Weights for the generative(augmentation) model $\leftarrow \theta$
Output: Augmented samples

repeat
  Take gradient descent step at
  $\nabla_\psi \left\| x - \mathtt{De}_\phi \left(\mathtt{En}_\psi(x, \mathcal{E}) \right) \right\|^2$
  $\nabla_\phi \left\| x - \mathtt{De}_\phi \left(\mathtt{En}_\psi(x, \mathcal{E}) \right) \right\|^2$
until Coverged
repeat
  $(x_0, c) \sim p(x, c)$
  $r_0 \leftarrow \mathtt{En}_\psi \left(x_0, \mathcal{E} \right)$
  $t \sim \mathrm{Uniform}(\{1, \ldots, T\})$
  $\epsilon \sim \mathcal{N}(0, \mathbf{I})$
  Take gradient descent step at
  $\nabla_\theta \left\| \epsilon - \epsilon_\theta \left(\sqrt{\bar{\alpha}_t} r_0 + \sqrt{1 - \bar{\alpha}_t} \epsilon, t, c \right) \right\|^2$
until Coverged
$r_T \sim \mathcal{N}(0, \mathbf{I})$
for $\mathrm{t} = \mathrm{T}, \ldots, 1$ **do**
  $z \sim \mathcal{N}(0, \mathbf{I})$ if $\mathrm{t} > 1$, else $\mathrm{z} = 0$
  $r_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(r_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}} \epsilon_\theta \left(r_t, t, c \right) \right) + \sigma_t z$
end for
return r_0
Augmented samples $\leftarrow \mathtt{De}_\phi(r_0)$

should be generated. In the conditional generation setting, the input data x_0 had an associated condition term (sensitive group and resistant group). Then, the diffusion model needed to be modified to include condition term c as an input to the reverse process for learning a conditioned generative model $p_\theta \left(x_0 | c \right)$.

$$p_\theta \left(x_{t-1} \mid x_t \right) = \mathcal{N} \left(x_{t-1}; \mu_\theta \left(x_t, t, c \right), \sigma_t^2 \boldsymbol{I} \right) \tag{8}$$

The reverse process, in which denoising occurs, is transformed into a conditional probability according to the given condition, and the injection of noise is the same for data belonging to any class, indicating that the forward process is not transformed based on the condition or class of the data. When the mean in the reverse process changes from $\mu_\theta \left(x_t, t \right)$ to $\mu_\theta \left(x_t, t, c \right)$, c should be added as an extra input to the trainable backbone architecture function approximators. In this case, depending on the modification of the reverse process, the original simplified loss function (Eq. (5)) can be rewritten as follows:

$$L_{\text{cond}} := \mathbb{E}_{t, \epsilon, x_0} \left[\left\| \epsilon - \epsilon_\theta \left(\sqrt{\bar{a}_t} x_0 + \sqrt{1 - \bar{a}_t} \epsilon, t, c \right) \right\|^2 \right] \tag{9}$$

To sample latent variables that encompass biological information under specific conditions, it is necessary to adjust the stochastic generation step, which has been redefined as ancestral sampling by Ho et al. [14] and Song et al. [30]. The objective is to modify the inference process to generate stochastic samples that align with the specified conditions. This modification can be described as follows:

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{a}_t}} \epsilon_\theta \left(x_t, t, c \right) \right) + \sigma_t z \tag{10}$$

Finally, we performed decoding from the generated latent space to the gene expression data via a trained graph AE model.

Section 4.3 describes the data augmentation method proposed for gene expression profiles to improve the generalization performance of drug response prediction models. This method aims to overcome the difficulty of generating gene expression data by using a compression(graph AE) model to map a low-dimension latent space from GE data that captures biological structure information and then generating that latent space with DDPM model; this method offers several advantages. By employing a graph AE to capture biological relationships during generative model training, the difficulty of model training is relatively reduced. Furthermore, mapping high-dimensional gene expression data to a low-dimensional latent space decreases computational complexity.

4.4 Drug Response Prediction with Graph Network

This section presents the final module of the proposed framework, which is a drug-response prediction method that employs graph attention networks (Figure 1.C). Specifically, each biological pathway was represented as a subgraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, and the graph attention network model was applied to each subgraph to learn the patterns of the gene relationships. Further details of this method are provided below.

4.4.1 Graph Attention Network. Each subgraph consisted of initial node features $X_i = H_{v_i}^{(0)} \in \mathbb{R}^{N \times F}$ and edges (v_i, v_j) describe the interaction between nodes. The node features comprised gene expression and indicators. Gene indicators were uniquely assigned to each gene using a trainable embedding matrix, resembling the process of token embedding in BERT [6]. Furthermore, nodes belonging to different biological pathways that corresponded to the same gene symbol shared the same gene-embedding space. The linkage information for each gene was extracted using the adjacency matrix obtained by preprocessing the data parsed from the Kyoto Encyclopedia of Genes and Genomes (KEGG) [17] database. Then, we adopted a graph learning method that employs the attention mechanism proposed by Ryu et al. [26] after the establishment of graph representation; this method is represented as follows:

$$H_{v_i}^{l+1,k} = \sigma \left(\sum_{v_j \in N(v_i)} \alpha_{i,j}^{l,k} H_{v_j}^{l,k} W^{l,k} \right), \mathcal{G}_k \in \mathbb{G} \tag{11}$$

$$\alpha_{i,j}^{l,k} = \sigma \left(\left(H_{v_i}^{l,k} W^{l,k} \right) C^{l,\mathcal{G}} \left(H_{v_j}^{l,k} W^{l,k} \right)^T \right) \tag{12}$$

where $N(v_i) = \{ v_j : (v_i, v_j) \in \mathcal{E} \}$ denotes the set of neighbors of $v_i \in \mathcal{V}$. k indicates index of set of subgraphs and $H_{v_i}^{l,k}$ denotes hidden vector of v_i node of l-th layer and k-th subgraph. And the i-th node state is updated as a function of the previous node state, $W^{l,\,k}$ are learnable parameters of the l-th layer, $\sigma(\bullet)$ is an activation function, $\alpha_{i,j}^{l,\,k}$ denotes an attention coefficient that measures the importance of the j-th node in updating the i-th state, and C represents the coupling matrix that combines the information of the i-th and j-th nodes in the graph. Regarding C, it is a learned parameter matrix that captures the pairwise relationship between nodes and calculates attention coefficients.

4.4.2 Readout, Concatenation, and Drug Response Predictions. After all the nodes in each subgraph were updated, a readout function was used to aggregate information of one subgraph. And concatenate operation integrated the distance information of the target protein for the biological pathway representing that subgraph with the information of the nodes in that subgraph (Eq. (13)).

$$Z_G = \mathrm{concat} \left(d_k^{-1} * \mathrm{MLP}_{\text{readout}} \left(H^{L,k} \right) \mid k = 0, ..., K \right) \tag{13}$$

where d_k represents the distance between the target protein and its subgraph (biological pathway), $H^{L,k}$ represents the final node state of k-th subgraph (The encoder layer of the graph autoencoder in Section 4.3.1 is also constructed as in Eq. (14)). After the information from all the networks was combined, it was concatenated to form a representation vector, which was then used to predict the final task of the drug response, as below.

$$y_{\text{pred}} = \text{MLP}_{\text{pred}}\ (Z_G) \tag{14}$$

There are two important differences between the method discussed in the current subsection and the model proposed by Ryu et al. First, among the features of the nodes that comprised each subgraph, the gene indicator was extracted from a shared trainable gene-embedding matrix, enabling networks to leverage shared information across all the biological pathways while still learning the specific characteristics of each pathway. In other words, by sharing the gene-embedding matrix, the networks could exploit the commonalities (identical genes) between different pathways while still learning their unique differences (geometric information). Second, by incorporating this distance information from the target protein into the readout function, the model could better capture the relationship between gene expression and drug response, thereby making more accurate predictions. Because genes that are closer to the target protein may have a greater effect on drug response than genes that are further away [16].

5 Experiments

This section presents the evaluation of the performance of the proposed framework in terms of its prediction performance for drug responses, generation performance for gene expression data. To verify these aspects, we conducted various experiments, which are described over various subsections.

Drug Response Prediction Performance (Sec 5.2): This subsection describes the improvement in the performance of drug response prediction using the proposed framework over that of the existing baseline model and analyzes any potential differences in performance between the complete framework and the framework without augmentation module.

Synthetic Data Quality (Sec 5.3): This section compares the data generated by the proposed model with those generated by other baseline generation models and discusses the effectiveness of the proposed model in capturing the distribution of real data.

Sensitivity Analysis (Sec 5.4): This section details the number of biological pathways selected during the process under the first module and the number of augmented data points needed in the second module for obtaining the best performance.

5.1 Dataset

We used the same dataset setup as that used in a previous study [27]; the training set consisted of samples composed of gene expression values and the corresponding drug response values for cell lines from the Genomics of Drug Sensitivity in Cancer (GDSC) database [34]. For the drug response labels (resistance/sensitive), we applied the experimentally determined cutoffs from previous study [16] to separate sensitive and resistant samples for each drug. The test set consisted of gene expression values and drug response information for patients from The Cancer Genome Atlas (TCGA) resource [31] and Patient-Derived Xenograft (PDX) encyclopedia [8]. The data were downloaded from the Zenodo repository [27]. To obtain information regarding gene interactions or connections, we utilized the Search Tool for the Retrieval of Interacting Genes/Proteins(STRING) [34] database to acquire Protein-Protein interaction (PPI) data, focusing on high-confidence links and selecting the largest connected component of the interactome. To ensure that only relevant interactions were considered, pathways and drugs with no genes in the PPI network were filtered out, resulting in 1,864 biological pathways.

5.2 Drug Response Prediction Performance

5.2.1 Experiment Setting. To train and validate the model, the GDSC dataset(train), PDX/TCGA(test) datasets were used for each of the six drugs. For a fairer comparison, we retained the training, test set construction, set of drugs, and evaluation metrics that we employed in our previous studies [25,29,30]. The training sets were divided at an 80:20 ratio to validate the trained models. We performed a stratified 5-fold cross-validation within the training set to determine the optimal number of epochs based on early stopping (training of the neural network was terminated when the validation loss stopped decreasing, with the patience value set to 15). and the optimal optimizer. The synthetic data obtained in section 4.3 were not included in the validation subsets during training, but only in the training subset. The data augmentation rate for the drug response prediction was set to 75% of the real sample. This ratio represented the generally ideal augmentation ratio obtained through the "searching optimal hyperparameter" experiment described later in section 5.4. For example, if the number of samples per label (resistant/sensitive) in the real data are n and m, respectively, the number of samples per label in the synthetic sample will be $\frac{(n+m)}{2}$ and $\frac{(n+m)}{2}$, respectively. The evaluation of the synthetic data is detailed in section 5.3. We measured the AUC values of the test set (PDX/TCGA) using the parameters with the highest average AUC values in the validation set.

5.2.2 Baselines. We evaluated the performance of several state-of-the-art methods for anticancer drug response prediction using multi-omics data. Specifically, three recently developed methods, MOLI [27], Super.FELT [23], and DeepInsight-3D [28], were used as benchmarks for comparison. In addition to these methods, non-negative matrix factorization (NMF) [21], feedforward net, and the method proposed by Geeleher et al. [9] were included in the comparison analysis. Additionally, a comparison was conducted with other methods mentioned in a previous study, including the autoencoder (AE) [7], artificial neural network after feature selection (ANNF) [23], AutoBoruta Random Forest (AutoBorutaRF) [33], and support vector machine (SVM) [15]. Finally, we compared the proposed framework with TabNet [2], a common model used in similar tasks with tabular data.

5.2.3 Performance Comparison. Table 1 presents an overview of the generalization performances of different baseline models and the proposed model for drug response predictions. The results revealed that the proposed DrDiff model obtained the highest AUC for all the drugs in the PDX benchmark dataset, except erlotinib, and the best performance for all the drugs on the TCGA benchmark dataset. Overall, the results suggested that the proposed method outperformed all the comparable baseline models, as evidenced by its higher average AUC of 0.79 for all six datasets. This finding suggested that the proposed method had a strong potential for improving the accuracy of drug response predictions, especially on the PDX and TCGA benchmark datasets. Starting from the problem definition of this studies that showed poor generalization performance due to lack of training data, we investigated the ability of our proposed framework to predict drug responses without the data augmentation module to ver-

ify whether it performs worse without it. Additionally, this experiment enables a practical performance comparison with other baseline models when utilizing the graph attention network using subnetworks (Section 4.3), another contribution proposed in this study, without data augmentation. Models trained solely on existing real data without augmentation exhibited lower performance for all six drugs compared to models trained with augmented data. This demonstrates that augmenting training data improves generalization performance on the benchmark dataset, which is unseen data, and suggests that the augmented training data is appropriately crafted to assist in the downstream task of drug response prediction. Moreover, utilizing only the graph attention network, the third module proposed in the DrDiff framework, without augmentation, achieved higher drug response prediction performance than all baseline models for two out of three drugs in the TCGA dataset. These experimental results highlighted that the second module (the data augmentation module) and third module (the graph attention network module) in the DrDiff framework were critical to the overall generalization performance.

Model \ Dataset	PDX set			TCGA set			**Avg**
	Pac	Cet	Erl	Doc	Cis	Gem	
NMF [21]	0.24	0.53	0.28	0.39	0.40	0.58	0.40
FFN [27]	0.68	0.43	0.37	0.69	0.44	0.65	0.54
Geeleher et al [9]	0.52	0.58	0.67	0.59	0.62	0.53	0.58
AE [7]	0.44	0.42	0.33	0.50	0.46	0.50	0.44
ANNF [23]	0.64	0.43	0.65	0.64	0.68	0.57	0.60
ABRF [33]	0.46	0.17	0.17	0.42	0.45	0.53	0.36
SVM [15]	0.49	0.41	0.67	0.53	0.47	0.47	0.50
TabNet [2]	0.51	0.51	0.58	0.50	0.50	0.50	0.51
MOLI [27]	0.74	0.53	0.63	0.58	0.66	0.65	0.63
Super.FELT [23]	0.64	0.55	0.76	0.64	0.73	0.61	0.65
DeepInsight [28]	0.74	0.71	**0.85**	0.78	0.68	0.53	0.71
DrDiff(No-Aug)	0.70	0.67	0.80	0.72	0.75	0.66	0.71
DrDiff	**0.78**	**0.82**	0.83	**0.83**	**0.81**	**0.72**	**0.78**

Table 1. Drug response prediction performance on the PDX and TCGA dataset. Pac, Cet, Erl, Doc, Cis, Gem respectively represent Paclitaxel, Cetuximab, Erlotinib, Docetaxel, Cisplatin, and Gemcitabine.

5.3 Synthetic Data Quality

Through a previous experiment (Section 5.2.3), we confirmed that adding the data generated by the proposed augmentation technique to the training data resulted in an improved prediction performance on the benchmark dataset. However, it was unclear whether this performance improvement was solely due to the high-quality data generated. Therefore, in this section, we discuss the verification of whether the data generated by the augmentation module properly mimicked the distribution of real data.

5.3.1 Experiment Setting. To conduct the experiment, we utilized the generative module with the hyperparameters discussed in section 5.2 to generate samples equivalent to that of the real samples in number. Then, for the quantitative evaluation of the sampling quality between that of the existing methods and proposed method, we used four evaluation metrics inspired by previous study [10] (Kullback–Leibler divergence (KLD), Pairwise Difference, Log-Cluster , and Cosine Similarity). The KLD was computed over a pair of real and synthetic marginal probability mass functions (PMFs) for a given variable, and the similarity between two PMFs was measured. The pairwise difference was measured as the Euclidean distance between each pair of real and synthetic data. The log-cluster metric was measured the similarity of the underlying latent structures of the real and synthetic datasets in terms of clustering. Cosine similarity was computed the similarity between the two nonzero vectors of an inner product space, and it measures the cosine of the angle between these vectors.

5.3.2 Baselines. We compared our augmentation module with four tabular data generative models, excepting the image data generative models. The conditional variational autoencoder (CVAE) [20] is a generative model that learns to generate new data samples by mapping a set of conditional variables to the output data distribution. This extends the traditional VAE by incorporating a set of conditional variables to control the generation process. We utilized a CVAE with the KL annealing technique inspired by β-VAE [13]. Conditional tabular GAN (CTGAN) [32] is another generative model that generates synthetic tabular data by learning the data distribution from real data and utilizing the generator to produce synthetic data that closely resemble the original data in terms of statistical properties. Tabular VAE (TVAE) [32] is a specialized VAE model designed for tabular data that aims to learn a compact latent representation of the data for downstream tasks such as data generation and anomaly detection. Finally, CopulaGAN [24] is a generative model implemented in the Synthetic Data Vault library that employs copulas to capture complex multivariate dependencies in tabular data. This approach enables the generation of synthetic data with a statistical structure resembling that of the original data.

	KLD(↓)	PD(↓)	Log-Cluster(↓)	Cosine Sim(↑)
CVAE [13]	0.0173	8.660	-1.389	0.972
CTGAN [32]	0.0124	6.076	**-1.521**	0.979
TVAE [32]	0.0068	4.848	-1.406	0.988
CopularGAN [24]	0.0159	7.088	-1.384	0.974
DrDiff	**0.0061**	**4.560**	-1.491	**0.989**

Table 2. Generation performances of different models

5.3.3 Quantitative Evaluation. Using the evaluation metrics mentioned in section 5.3.1, we evaluated the quality of the generated data for real gene expression data for each of the six drugs (Table 2). Subsequently, we derived the final performance of model by calculating the average results of the four evaluation metrics for each drug. When comparing final performance under the four performance indicators for each model with the four baseline models and the proposed modules, our proposed model achieved a significantly superior performance in terms of the KLD, pairwise distance, and cosine similarity metrics, except for the log-cluster metrics. This observation demonstrates that the proposed adeptly approximates the distribution of real-world data and effectively captures the interrelationships among features.

5.3.4 Qualitative Evaluation. For the qualitative evaluation of the sampling diversity between the baseline generative model used in the quantitative evaluation and our proposed augmentation module, we conducted a t-SNE visualization comparison analysis between real and synthetic data (Figure 2). When compared with the t-SNE distribution of the real data, the data generated by the CTGAN and CopulaGAN models exhibited a significant difference in distribution owing to the excessive inclusion of nonexistent values. In contrast, the CVAE model generated data within the range of the actual data; however, its diversity was limited. The proposed augmentation module outperformed the other baseline models in terms of sampling diversity and showed an almost identical distribution to that of real

data.

5.3.5 Controllable Generation. To demonstrate that our proposed augmented module could generate a label given as a condition, training data was augmented for both [matched] and [unmatched] cases, as shown in Figure 3(a). The augmentation rate was set to 75% of the real data, which is the training data augmentation rate used in section 5.2. Figure 3(b) shows the validation loss when training the drug response prediction model for the two cases with augmented training data for docetaxel. The decrease in the validation loss for the [unmatched] cases was smaller than that for the [matched] cases. This suggested that the prediction model did not learn well for drug response predictions when trained on the [unmatched] cases. In other words, our proposed generation model could demonstrate to be capable of controlling drug response prediction for a given condition (resistance/sensitivity).

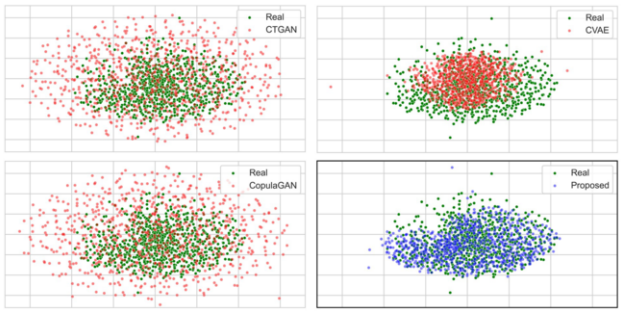


Figure 2. t-SNE visualizations of the synthetic and real data

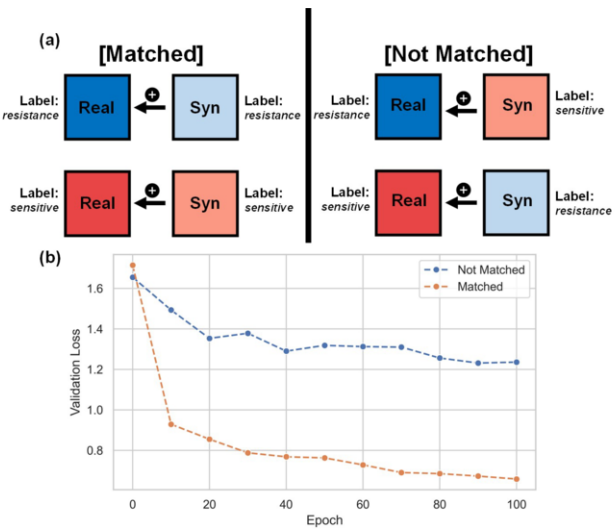


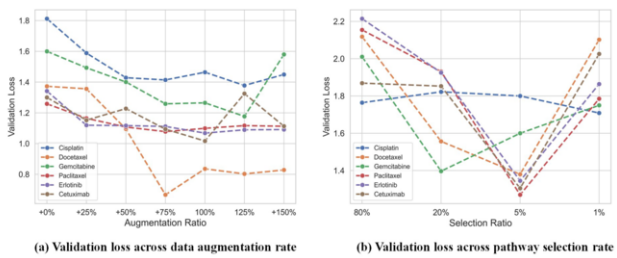
Figure 3. Augmentation of training data for matched and unmatched cases

5.4 *Sensitivity Analysis*

In this section, we describe the experiments conducted to determine when the generalization performance of the drug response prediction model was optimal by assessing the number of augmented training data and the number of selected biological pathways.

5.4.1 Experiment Setting. The experimental setup closely resembled that described in section 5.2.1. The training set was divided in a ratio of 80:20 for model validation, with synthetic data excluded from the validation set. Notably, the independent benchmark datasets PDX and TCGA were not considered in this study. Additionally, a stratified 5-fold cross-validation was performed within the training set at a ratio of 8:2 for training and validation. The average of the recorded final validation losses for each fold was defined as the performance metric under the given experimental conditions.

5.4.2 Performance. Figure 4 shows the pattern of the average final cross-validation loss values depending on the degree of training data augmentation (Section 4.1) and the degree of biological pathway selection (Section 4.2). The performance improvement concerning the augmentation and selection degrees was not consistent pattern across the six drug cases. However, a common finding across all drug cases is that there is a threshold for performance enhancement owing to training data augmentation. In other words, infinitely increasing the data did not lead to unlimited improvements in the performance of the downstream task (drug response prediction). Similarly, we confirmed the presence of a threshold for performance improvement due to pathway selection. Additionally, we found that selecting more pathways did not lead to an infinite performance improvement. The optimal drug response prediction performance is generally achieved when the data augmentation ratio ranges from 75% to 125% of the original data, as observed across all drugs. To perform the drug response prediction task, the optimal data augmentation ratio was determined to be 75%, as it yielded the lowest average validation loss for all the drugs across different ratios (25, 50, 75, 100, 125, and 150%). Similarly, the optimal hyperparameter for the pathway selection ratio was determined by selecting only 5% of the total pathways, following the previously mentioned method.



(a) Validation loss across data augmentation rate (b) Validation loss across pathway selection rate

Figure 4. Validation loss compared to the augmentation ratio and selection ratio

6 Conclusion

In this study, we propose a drug response prediction framework that overcomes the low sample size and high dimensionality of training data. To solve this problem, we performed feature selection to extract only important variables through the distance between a drug and target protein; employed data augmentation to generate gene expression data by modifying the latest generation model, DDPM; and utilized a graph attention network for drug response prediction using biological pathways as the a priori knowledge. The proposed framework showed a higher generalization performance on unseen patient datasets for drug response prediction than the performances of other baselines. Furthermore, the generation module that augmented the gene expression data also produced a higher sample quality than that produced by the other comparison models.

Acknowledgements

This research was supported by the National Research Foundation (NRF) funded by the Korean government (MSIT) (No. RS-2023-00229822). The funders did not play any role in the design of the study, data collection, analysis, or preparation of the manuscript.

References

[1] A. Al-Mekhlafi, T. Becker, and F. Klawonn. Sample size and performance estimation for biomarker combinations based on pilot studies with small sample sizes. *Communications in Statistics-Theory and Methods*, 51(16):5534–5548, 2022.

[2] S. Ö. Arik and T. Pfister. Tabnet: Attentive interpretable tabular learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 6679–6687, 2021.

[3] P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner, et al. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018.

[4] C. Chadebec and S. Allassonnière. Data augmentation with variational autoencoders and manifold sampling. In *Deep Generative Models, and Data Augmentation, Labelling, and Imperfections: First Workshop, DGM4MICCAI 2021, and First Workshop, DALI 2021, Held in Conjunction with MICCAI 2021, Strasbourg, France, October 1, 2021, Proceedings 1*, pages 184–192. Springer, 2021.

[5] P. Chaudhari, H. Agrawal, and K. Kotecha. Data augmentation using mg-gan for improved cancer classification on gene expression data. *Soft Computing*, 24:11381–11391, 2020.

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

[7] M. Q. Ding, L. Chen, G. F. Cooper, J. D. Young, and X. Lu. Precision oncology beyond targeted therapy: combining omics data with machine learning matches the majority of cancer cells to effective therapeutics. *Molecular cancer research*, 16(2):269–278, 2018.

[8] H. Gao, J. M. Korn, S. Ferretti, J. E. Monahan, Y. Wang, M. Singh, C. Zhang, C. Schnell, G. Yang, Y. Zhang, et al. High-throughput screening using patient-derived tumor xenografts to predict clinical trial drug response. *Nature medicine*, 21(11):1318–1325, 2015.

[9] P. Geeleher, N. Cox, and R. Huang. Clinical drug response can be predicted using baseline gene expression levels and. *Vitro Drug sensitivity Cell lines. Genome Biol*, 15:R47, 2014.

[10] A. Goncalves, P. Ray, B. Soper, J. Stevens, L. Coyle, and A. P. Sales. Generation and evaluation of synthetic patient data. *BMC medical research methodology*, 20:1–40, 2020.

[11] E. Guney, J. Menche, M. Vidal, and A.-L. Barábasi. Network-based in silico drug efficacy screening. *Nature communications*, 7(1):10331, 2016.

[12] W. Hamilton, Z. Ying, and J. Leskovec. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30, 2017.

[13] I. Higgins, L. Matthey, A. Pal, C. P. Burgess, X. Glorot, M. M. Botvinick, S. Mohamed, and A. Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. *ICLR (Poster)*, 3, 2017.

[14] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[15] C. Huang, R. Mezencev, J. F. McDonald, and F. Vannberg. Open source machine-learning algorithms for the prediction of optimal cancer drug therapies. *PLoS One*, 12(10):e0186906, 2017.

[16] F. Iorio, T. A. Knijnenburg, D. J. Vis, G. R. Bignell, M. P. Menden, M. Schubert, N. Aben, E. Gonçalves, S. Barthorpe, H. Lightfoot, et al. A landscape of pharmacogenomic interactions in cancer. *Cell*, 166(3): 740–754, 2016.

[17] M. Kanehisa and S. Goto. Kegg: kyoto encyclopedia of genes and genomes. *Nucleic acids research*, 28(1):27–30, 2000.

[18] T. Karras, M. Aittala, J. Hellsten, S. Laine, J. Lehtinen, and T. Aila. Training generative adversarial networks with limited data. *Advances in neural information processing systems*, 33:12104–12114, 2020.

[19] S. Kim, S. Bae, Y. Piao, and K. Jo. Graph convolutional network for drug response prediction using gene expression data. *Mathematics*, 9 (7):772, 2021.

[20] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[21] D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. *Advances in neural information processing systems*, 13, 2000.

[22] B. Liu, Y. Wei, Y. Zhang, and Q. Yang. Deep neural networks for high dimension, low sample size data. In *IJCAI*, volume 2017, pages 2287–2293, 2017.

[23] S. Park, J. Soh, and H. Lee. Super. felt: supervised feature extraction learning using triplet loss for drug response prediction with multi-omics data. *BMC bioinformatics*, 22(1):269, 2021.

[24] N. Patki, R. Wedge, and K. Veeramachaneni. The synthetic data vault. In *2016 IEEE international conference on data science and advanced analytics (DSAA)*, pages 399–410. IEEE, 2016.

[25] D. M. Roden, R. A. Wilke, H. K. Kroemer, and C. M. Stein. Pharmacogenomics: the genetics of variable drug responses. *Circulation*, 123 (15):1661–1670, 2011.

[26] S. Ryu, J. Lim, S. H. Hong, and W. Y. Kim. Deeply learning molecular structure-property relationships using attention-and gate-augmented graph convolutional network. *arXiv preprint arXiv:1805.10988*, 2018.

[27] H. Sharifi-Noghabi, O. Zolotareva, C. C. Collins, and M. Ester. Moli: multi-omics late integration with deep neural networks for drug response prediction. *Bioinformatics*, 35(14):i501–i509, 2019.

[28] A. Sharma, A. Lysenko, K. A. Boroevich, and T. Tsunoda. Deepinsight-3d architecture for anti-cancer drug response prediction with deep-learning on multi-omics. *Scientific reports*, 13(1):2483, 2023.

[29] V. Solhaug and E. Molden. Individual variability in clinical effect and tolerability of opioid analgesics–importance of drug interactions and pharmacogenetics. *Scandinavian journal of pain*, 17(1):193–200, 2017.

[30] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[31] J. N. Weinstein, E. A. Collisson, G. B. Mills, K. R. Shaw, B. A. Ozenberger, K. Ellrott, I. Shmulevich, C. Sander, and J. M. Stuart. The cancer genome atlas pan-cancer analysis project. *Nature genetics*, 45(10): 1113–1120, 2013.

[32] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni. Modeling tabular data using conditional gan. *Advances in neural information processing systems*, 32, 2019.

[33] X. Xu, H. Gu, Y. Wang, J. Wang, and P. Qin. Autoencoder based feature selection method for classification of anticancer drug response. *Frontiers in genetics*, 10:433119, 2019.

[34] W. Yang, J. Soares, P. Greninger, E. J. Edelman, H. Lightfoot, S. Forbes, N. Bindal, D. Beare, J. A. Smith, I. R. Thompson, et al. Genomics of drug sensitivity in cancer (gdsc): a resource for therapeutic biomarker discovery in cancer cells. *Nucleic acids research*, 41(D1):D955–D961, 2012.