GEFA: Early Fusion Approach in Drug-Target Affinity Prediction

Tri Minh Nguyen, Thin Nguyen, Thao Minh Le, and Truyen Tran
Applied Artificial Intelligence Institute
Deakin University
Victoria, Australia
(minhtri, thin.nguyen, lethao, truyen.tran)@deakin.edu.au

Abstract

Predicting the interaction between a compound and a target is crucial for rapid drug repurposing. Deep learning has been successfully applied in drug-target affinity (DTA) problem. However, previous deep learning-based methods ignore modeling the direct interactions between drug and protein residues. This would lead to inaccurate learning of target representation which may change due to the drug binding effects. In addition, previous DTA methods learn protein representation solely based on a small number of protein sequences in DTA datasets while neglecting the use of proteins outside of the DTA datasets. We propose GEFA (Graph Early Fusion Affinity), a novel graph-in-graph neural network with attention mechanism to address the changes in target representation because of the binding effects. Specifically, a drug is modeled as a graph of atoms, which then serves as a node in a larger graph of residues-drug complex. The resulting model is an expressive deep nested graph neural network. We also use pre-trained protein representation powered by the recent effort of learning contextualized protein representation. The experiments are conducted under different settings to evaluate scenarios such as novel drugs or targets. The results demonstrate the effectiveness of the pre-trained protein embedding and the advantages our GEFA in modeling the nested graph for drug-target interaction.

1 Introduction

Predicting drug-target binding affinity (DTA prediction) is crucial in new drug development as well as drug repurposing [1]. Machine learning methods are particularly attractive because they offer cheap and fast alternatives with reasonable performance thanks to the large DTA databases [2] that we can leverage on. With the advance of machine learning, many computational prediction methods [3]–[6] have been proposed to tackling DTA. In recent works, the protein is typically represented as amino acids sequence [5]–[7]. The drawback of using protein sequence is that it can not represent the 3D structure. However, obtaining the high-resolution 3D structure is a challenging task. A more practical solution is using the 2D residue contact maps to represent tertiary protein structure. The contact map can be naturally modeled using recent advances in deep learning known as graph neural networks (GNN). Here each residue is represented as a node in the graph, and a contact between two residues as an edge.

Previous deep learning-based DTA prediction methods [5]–[8] often use the late fusion approach. The late fusion approach extracts drug and target representation separately then predicts the binding affinity from the combined representation at the very end of the process. However, this practice ignores the fact that the binding occurs at a pocket rather than the whole protein. Once the drug binds, it changes the protein functions to have pharmaceutical effects, hence it can also change the

Machine Learning for Structural Biology Workshop, NeurIPS 2020, Vancouver, Canada.

protein structure [9], hence its representation. The change in protein representation due to the binding process is not addressed in late fusion approach. In addition, the model assumes non site-specific binding, making it difficult to assign the credit to the sites that interact. It can also result in slower learning rate, and less interpretable prediction.

To address target protein representation change, we propose an early-fusion-based approach. Initially, we extract representation feature for a given drug molecule from its drug graph structure. Then, the drug representation is integrated into the protein graph structure before the protein representation learning phrase. This is basically a graph structure nested inside another graph structure. This graph-in-graph neural network design allows the model to learn changes in protein representation caused by the binding process with the drug molecule.

Previous works [5]–[7], [10] normally use one-hot encoding to vectorize residues. This conventional approach fails to embed the contextual dependencies between residues as well as not being able to make use of unlabeled protein sequences. We take advantage of the power of the protein embedding features learned by a protein language modeling on a large collection of protein sequences, including proteins that are not available in DTA datasets, to represent the residues in a given target protein. In this work, we refer target proteins whose the binding affinities available in the DTA datasets as labeled ones while proteins that do not exist in the DTA datasets as unlabeled proteins.

In summary, the contribution of our work is two-fold. First, we combine the protein sequence embedding feature and protein contact map to build the graph representation of a target protein. Second, in order to reflect the target representation change during the binding process, we propose a so-called Graph Early Fusion for binding Affinity prediction (GEFA) for more accurate biological modeling. We demonstrate the effects of the GEFA on Davis dataset [11] where it has shown superior performance against previous studies on different settings. Our Python implementation is publicly available at `https://github.com/ngminhtri0394/GEFA`.

2 Related Works

2.1 Drug-Target Binding Affinity Prediction Problem

Drug-target binding affinity indicates the strength of the binding force between the target protein and its ligand (drug or inhibitor) [12]. The drug-target binding affinity prediction problem is a regression task predicting the value of the binding force. The binding strength is measured by the equilibrium dissociation constant (K_D). A smaller K_D value indicates a stronger binding affinity between protein and ligand [12]. There are two main approaches: structural approach and non-structural approach [1]. Structural methods utilize the 3D structure of protein and ligands to run the interaction simulation between protein and ligand. On the other hand, the non-structural approach relies on ligand and protein features such as sequence, hydrophobic, similarity or other alternative structural information.

2.2 Non-structural Approach

The non-structural approach solves the binding affinity regression task without the accurate 3D structure of the target. Instead of using the 3D coordinate of target and drug atom. The non-structural approach relies on the drug-drug, target-target similarity, target and drug atom sequence, and other alternative structural information such as contact map or secondary structure.

2.2.1 Deep Learning Based Methods

DeepDTA [5] predicts the binding affinity from the 1D representation of protein and drug. WideDTA [7] is an extension of DeepDTA. Protein is represented not only in sequence but also in motif and domain. The drug is represented in SMILES and Ligand Maximum Common Substructures. Instead of using 1D representation for drug, GraphDTA [6] uses graph to express the interaction between atoms of the molecules. This allows modeling the interaction between any two atoms within the drug molecules. DrugVQA [8] uses distance map to represent the protein. Sequential self-attention is used to learn which parts of the protein interact with the ligand. Multi-head self-attention is used to learn which atoms in drugs have high contribution to the drug-target interaction. Graph-CNN [13] pretrains the protein pocket graph autoencoder by minimizing representation difference. The binding interaction model has protein pocket graph and 2D molecular graph as the inputs. The unsupervised learning helps the model to overcome the limited pocket graph training data. DGraphDTA [10] uses contact map to build protein graph structure with PSSM, one-hot encoding, and residue properties as node features. This allows the model to obtain an accurate protein representation.

3 Proposed Methods

The task of drug-target binding affinity (DTA) problem is to predict the binding affinity A between a target protein P and drug compound D. Mathematically, the problem is formulated as a regression task:

$$A = \mathcal{F}_\theta(P, D), \tag{1}$$

where θ is model parameters of predicting function $\mathcal{F}$.

In this section, we present details of our approach to solve DTA. In Sec. 3.1 explains the feature representation of target protein P, followed by the feature representation of drug compound D in Sec. 3.2. The main contribution of ours is presented in Sec. 3.3 where we aim at reflecting the changes in the target protein representation due to the conformation change.

3.1 Graph Representation of Protein

Given that we have the contact map information, each target protein is considered as a graph structure where nodes are residues in the protein sequence. Leveraging the rich set of unlabeled proteins, we utilize state of the art language modeling methods to learn contextual residue embedding representation. Language modeling is learning the representation of sequence using pretext tasks such as predicting missing token or the next token in sequence [14]. TAPE is a variant of language model for protein representation in particular. We use embedded representation learnt from a large collection of unlabeled protein sequences provided by TAPE [15] instead of one-hot encoding to represent each node in the protein graph. Subsequently, given a protein sequence of L residues, the node features of the protein graph is a set $\mathcal{V}_p = \left\{v_i \mid v_i \in R^h\right\}_{i=1}^{L}$, where h is the length of the embedding vector v_i provided by TAPE. Each v_i is contextual, that is residues occur in the context of surrounding residues. Therefore, the structural information is implicitly encoded into the embedding.

We use secondary structure as it decides the backbone shape of the target protein which also contributes to the shape of the binding site and overall structure. For each residue, the secondary structure feature is represented as the probability of three secondary structure type α helix, β pleated sheet, and coil.

Solvent accessibility indicates the level of interaction between residues and drug molecule. Solvent accessibility is divided into three classes: buried (pACC from 0 to 10), medium (pACC from 11 to 40), and exposed (pACC from 41 to 100). Eventually, the combination of embedding vector extracted by TAPE, secondary structure feature vector, and solvent accessibility feature vector are used to represent node features of residues in a target protein graph.

The contact map information provides the contacts between any two residue nodes in a protein graph. The sequence information is also retained in the graph structure in form of edges linking any two nodes of adjacency residues in the protein sequence. In practice, the contact map and sequence information are stored as an adjacency matrix $\mathcal{A}_p$. In the rest of this paper, we denote the protein graph as $\mathcal{G}_p = (\mathcal{V}_p, \mathcal{A}_p)$, where $\mathcal{V}_p$ are residue nodes in the protein chain.

3.2 Graph Representation of Drugs Compounds

The input drug compound is in the SMILES format. In the graph representation of molecule, atoms are nodes while the bonds between atoms are edges. The node feature consists of five properties: atom symbol, atom degree which is the total number of bonded atom neighbors, the number of hydrogens, implicit value of the atom, and whether if the atom is aromatic. These features are concatenated to form a multi-dimensional feature. The edges are expressed by an adjacency list indicating if there are bonds between any two atoms in the compounds. As the bonds are symmetric, a drug compound graph is a bidirectional graph. In the later use of graph representation of a drug compound, we refer it as $\mathcal{G}_d = (\mathcal{V}_d, \mathcal{A}_d)$, where $\mathcal{V}_d$ is atom features and $\mathcal{A}_d$ is bonds between atoms.

3.3 Graph Early Fusion for binding Affinity prediction (GEFA)

The overall architecture of our proposed method is presented in Fig.1. Our GEFA takes as input the graph structure of drug $\mathcal{G}_d$ and the graph structure of target $\mathcal{G}_p$ and outputs the prediction of binding affinity. We use Graph Convolutional Network (GCN) [16] for graph representation. In addition, we also make use of the well-known residual skip connection trick to make use of very deep GCNs.

3.3.1 Graph Convolutional Network

GCN is a convolutional network designed specifically for graph-structured signals. The goal is learn the node-level representation from a given input graph $\mathcal{G} = (\mathcal{X}, \mathcal{A})$ where $\mathcal{X}$ is node feature matrix of N nodes and $\mathcal{A} \in R^{N \times N}$ is the adjacency matrix that describes the graph structure. Let W^l be the weight matrix at l-th layer, the graph convolution operation is defined by:

$$H^1 = \mathcal{X}, \tag{2}$$

$$H^l = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{\mathcal{A}}\tilde{D}^{-\frac{1}{2}}H^{l-1}W^{l-1}\right), \tag{3}$$

where $\tilde{\mathcal{A}} = \mathcal{A} + \mathcal{I}$ is the adjacency matrix with self-loop in each node. $\mathcal{I}$ is the identity matrix and $\tilde{D} = \sum_j \tilde{\mathcal{A}}_{ij}$ and σ is a non-linear function which is a ReLU [17] in our later experiments.

3.3.2 Deeper GCN with Residual Blocks

In general, a deeper model can generalize better and more compact than shallow networks[18]. However, stacking the vanilla GCN often suffers from the problem of gradient vanishing and numerical instability as a consequence of matrix multiplication in Eq. 3. To mitigate this problem, we use the GCN with residual skip-connection proposed in [19]. Similar to the effect of the residual block in the well-known CNN [20], skip connection in GCN helps to create more direct gradient flow, hence, allows to go deeper with more convolution layers. Mathematically, the graph convolution operation is given by:

$$H^1 = \mathcal{X}, \tag{4}$$

$$F^l(H^{l-1}) = W_2^{l-1}\sigma(\mathcal{A}H^{l-1}W_1^{l-1} + b^{l-1}), \tag{5}$$

$$H^l = \sigma(H^{l-1} + F^l(H^{l-1})), \tag{6}$$

where W_1, W_2 are learnable weight matrices, l is layer-wise index and σ is a non-linear function which is a ReLU activation function.

3.3.3 Graph-Graph Integration with Early Fusion

To reflect the changes in representation of a target protein due to the interaction between drug molecule and protein, we propose Graph Early Fusion for binding Affinity (GEFA), a method for migrating the drug molecule graph $\mathcal{G}_d = (\mathcal{V}_d, \mathcal{A}_d)$ into the protein graph $\mathcal{G}_p = (\mathcal{V}_p, \mathcal{A}_p)$ via a self-attention mechanism.

We first refine node representations in the drug graph with a two-layers GCN as in Eq. 3 and residual blocks as in Eq. 4, 5, 6. Let $\mathcal{V}'_d = \{v'_i \mid v'_i \in R^{h_1}\}_{i=1}^{S}$ as node features of the drug graph after GCN, where S is number of nodes in the drug graph. Note that v'_i contains aggregating information from its neighbors so we simply use the largest estimated representation of the refined drug graph as the representation of the entire graph. This is easily obtained by a max pooling operation followed by two linear layers for feature projection:

$$v'_{\text{max}} = \text{MaxPool}(\mathcal{V}'_d), \tag{7}$$

$$x_d = (W_0 v'_{\text{max}} + b_0)W_1 + b_1. \tag{8}$$

We call the resulted vector $x_d \in R^{h_1}$ as the drug molecules node, where h_1 is dimension of x_d.

We now explain how we integrate the drug molecules node x_d into the protein graph $\mathcal{G}_p = (\mathcal{V}_p, \mathcal{A}_p)$ which is the main contribution of our work. The key idea is to use the drug node x_d as an additional node that binds to the target graph $\mathcal{G}_p$. The edges connecting the drug node and residue nodes in the protein graph indicate the interaction between residues and drug molecule as well as the binding site. Since not every residue contributes equally to the binding affinity, the edge weights indicate the level of interactions of each residue with the drug molecule. To learn the level of contribution, we utilize a self-attention mechanism driven by the residue features $\mathcal{V}_p = \{v_i \mid v_i \in R^{h_2}\}_{i=1}^{L}$, recalling that L is the length of the protein sequence. The self-attention mechanism is motivated from the fact that the binding site of the protein depends on the protein structure. In the other word, the attention weights tell which residues are more likely to participate in the binding process. Mathematically, the attention weights are given by:

$$\alpha_i = \text{sotfmax}(W_2 \text{tanh}(W_1 v_i)), \tag{9}$$

where $\sum_{i=1}^{L} \alpha_i = 1, v_i$ is the i-th residue feature, and W_1 and W_2 are the learnable parameters.

Given the drug node x_d and its connections to residues in the target protein graph $\mathcal{G}_p$ denoted by $\{\alpha_i\}_{i=1}^{L}$, we now construct a cross-domain graph $\mathcal{G}_{pd} = \{\mathcal{V}_{pd}, \mathcal{A}_{pd}\}$ where $\mathcal{V}_{pd} = \{\mathcal{V}_p, x_d\}$ and $\mathcal{A}_{pd} = \{\mathcal{A}_p, \{\alpha_i\}_{i=1}^{L}\}$. Similar to what we have done with the drug graph earlier, we employ a two-layers GCN followed by residual blocks to refine the node representations of the drug-protein graph $\mathcal{G}_{pd}$.

Before performing the graph feature extraction, the drug node after fusion v'_{dp} in the refined nodes $\mathcal{V}'_{pd}$ by GCNs is taken out from the protein graph to ensure that the graph feature only contains residues nodes. Eventually, we extract the latent representation of the protein graph with a global max pooling operator followed by a two-layer linear network:

$$v'_{\text{max}} = \text{MaxPool}(\mathcal{V}'_p), \tag{10}$$

$$v_{pd} = (W_0 v'_{\text{max}} + b_0) W_1 + b_1, \tag{11}$$

where $\mathcal{V}'_p = \{\mathcal{V}'_{pd} \setminus v'_{dp}\}$ is the node representations of the protein graph after removing the drug node.

At the same time, the drug latent vector x_d is transformed into the same dimensional space with v'_{dp} via a linear transformation. We further obtain the final representation of drug by combining these two features with a simple concatenation operator.

$$v_{db} = [x_d; v'_{dp}], \tag{12}$$

where [;] denotes the concatenation operation of two vectors. A max-pooling operation is then performed along the channel dimension to obtain the combined drug representation.

$$v_{dc} = \text{MaxPool}(v_{db}). \tag{13}$$

Consequently, the drug vector v_{dc} and the protein latent vector v_{pd} are concatenated and finally fed into a predictor of three fully connected layers to predict the binding affinities.

We wish to hypothesize that our early fusion approach with self-attention has two benefits. First, the early fusion approach explicitly models interactions between the drug graph and the target protein graph. Second, the self-attention allows the learning model to be more interpretable by showing which residues interact with drug molecules and how much they contribute to the binding process. We will back these in our later experiments.

4 Experiments

We evaluate our proposed model GEFA on Davis dataset[11] and compare against a late fusion baseline as well as state-of-the-art methods including GCNConvNet [6], GINConvNet [6], DGraphDTA [10]. Among those methods, GCNConvNet and GINConvNet use protein sequence and drug molecule graph as the input while DGraphDTA uses a protein graph built from contact and drug molecule as the input. We present the qualitative results in Sec. 4.1 and further provide analysis of our proposed model via extensive ablation studies in Sec.4.2.

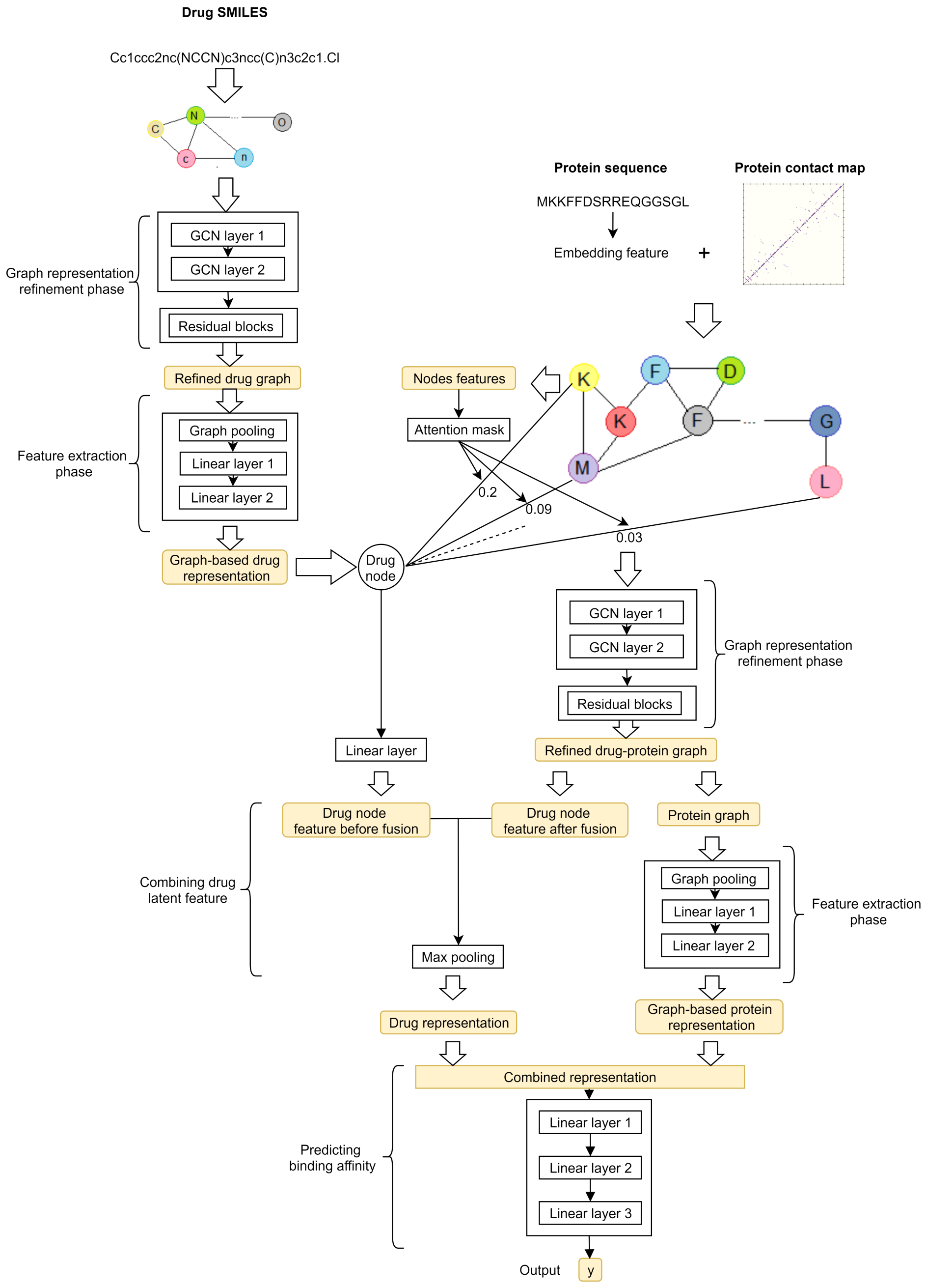

Figure 1: Illustration of Graph Early Fusion for binding Affinity prediction (GEFA). The GEFA takes as input the graph representations of a drug molecule and a protein target. We first use graph convolution network (GCN) for the drug graph feature refinement before applying a max pooling operator to obtain the estimated representation of the entire drug graph. The drug estimated vector is then used as an additional node to the protein graph, establishing graph-in-graph representation across domains. Information retrieved from the drug-protein graph along with the drug representation are finally used as the input for predicting the binding affinities between the given drug molecule and the protein target.

4.1 Quantitative Experiments

4.1.1 Dataset

Davis dataset consists of binding affinity information between 72 drugs and 442 targets. The binding affinity between drug and target is measured by K_D (kinase dissociation constant) value [11]. For the Davis dataset, the drug SMILES sequence of 68 drugs and the target protein sequence of 442 targets from DeepDTA[5] training/test set are used in our experiments.

There are four experiments settings for four scenarios. The first experiment setting is the warm setting where both protein and drug are known to the model. The second experiment is cold-target where proteins are unknown to the model and drugs are known to the model. In this case, each unique protein sequence only appears in training, validation, or test set. As targets in the cold-target setting are required to be unique in both train, validation, and test sets, targets having the same sequence are filtered out which results in 361 targets. The 361 targets are split at 0.8/0.2 ratio for training-validation/testing. Then the training set is split at 0.8/0.2 ratio for training/validation. The third experiment setting is cold-drug where proteins are known to the model and drugs are unknown to the model. We conduct the same splitting procedure in cold-target but applying for drugs. Finally, the last experiment setting is cold-drug-target where both drugs and proteins are unknown to the model. In this setting, we conduct the splitting procedure for both drug and target to ensure training, validation, and testing set do not share any common drug or target.

4.1.2 Implementation Details

Our methods are implemented using Pytorch. The protein sequence embedding features are extracted using TAPE-Protein [21]. The contact map is predicted by RaptorX [22]. The TAPE-Protein uses BERT language modelling [14]. The output of TAPE-Protein embedding features extraction is a embedding vector size 768. The graph convolution network uses the Pytorch geometric library [23]. The model are trained on 128 mini-batch. The learning rate is 0.0005 in warm setting. In cold-target, cold-drug, and cold-drug-target setting, the learning rate is 0.001 as higher learning rate helps model to have better generalization and less likely to overfit. The learning rate decay is used. The learning rate is reduced by 20% every 40 epochs without improvement in MSE metric in the validation set. Adam optimizer is used. The model is trained in 1000 epochs.

4.1.3 Evaluation Metrics

The models' performances are evaluated using Concordance Index (CI)[24], Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Pearson[25], and Spearman[26].

4.1.4 Results

To compare our early fusion approach with the conventional late fusion approach, we provide a late fusion baseline model. Our late fusion baseline model (GLFA - Graph Late Fusion for binding Affinity) follows the convention model in which drug and protein representation are learned separately. GLFA has graph structures of drug and target as the input. Both graph structures are processed using the two-layers GCN and residual blocks in parallel to learn the hidden features. Then, the latent features are obtained by global max pooling followed by two linear layers. The latent features from both protein and drug are concatenated before under-going three fully connected layers. The output of the final fully connected layers is the binding affinity value of the input drug and target protein. The late fusion model is a special case of early fusion where all drug-residues edge weights are set to 0. The late fusion approach is used as baseline to compare with our proposed early fusion approach.

We report our late fusion approach, GLFA, and early fusion approach, GEFA, with previous works in Davis benchmark on four settings in Table 1. Our proposed method GEFA consistently outperforms previous works in four settings. Our proposed methods achieve state-of-the-art performance across all four settings. Between two late fusion based methods DGraphDTA [10] and GLFA, our proposed GLFA method also outperforms DGraphDTA. This follows our expectations as the embedding feature contains richer information than one-hot encoding and PSSM. This also demonstrates the advantage of using the residual block.

DgraphDTA [10], GLFA, and GEFA outperform GINConvNet in all four settings. GINConvNet and GCNConvNet [6] only use sequence and CNN to learn the target representation. On the other hand,

Table 1: The result of quantitative experiments

Architecture	RMSE↓	MSE↓	Pearson↑	Spearman↑	CI ↑
Warm start setting					
GCNConvNet [6]	0.5331	0.2842	0.8043	0.6609	0.8649
GINConvNet [6]	0.50723	0.2573	0.8245	0.6818	0.8785
DGraphDTA [10]	0.4917	0.2417	0.8378	0.7001	0.8869
GLFA	0.4850	0.2353	0.8412	**0.7073**	**0.8950**
GEFA	**0.4775**	**0.2280**	**0.8467**	0.7023	0.8927
Cold-target setting					
GCNConvNet [6]	0.7071	0.5000	0.5145	0.4316	0.7293
GINConvNet [6]	0.7144	0.5104	0.5166	0.3904	0.7065
DGraphDTA [10]	0.6855	0.4700	0.5597	0.4941	0.7656
GLFA	0.6732	0.4531	0.5828	0.5228	0.7802
GEFA	**0.6584**	**0.4335**	**0.6030**	**0.5506**	**0.7951**
Cold-drug setting					
GCNConvNet [6]	0.9723	0.9454	0.3385	0.3764	0.6784
GINConvNet [6]	0.9592	0.9200	0.3779	0.3693	0.6758
DGraphDTA [10]	0.9583	0.9184	0.3610	0.3150	0.5337
GLFA	0.9280	0.8612	0.4023	0.3549	0.6703
GEFA	**0.9202**	**0.8467**	**0.4515**	**0.4320**	**0.7091**
Cold-drug-target setting					
GCNConvNet [6]	1.0632	1.1304	0.1904	0.1698	0.5782
GINConvNet [6]	1.0651	1.1345	0.1974	0.2763	0.6275
DGraphDTA [10]	1.0749	1.1554	0.0228	0.1795	0.6081
GLFA	1.0698	1.1444	**0.3473**	0.2901	0.6362
GEFA	**0.9949**	**0.9899**	0.3148	**0.2932**	**0.6390**

DgraphDTA [10], GLFA, and GEFA use the graph built from the protein contact map and learn the target representation using GCN. This demonstrates the advantage of using the graph representation of the contact map.

Between our two proposed methods GEFA and GLFA, the early fusion method GEFA shows advantages over late fusion method GLFA. This follows our expectation as the early fusion allows interactions between drug and protein graph during the graph representation learning phase for more accurate latent representation.

4.2 Ablation Studies

To understand the contribution of each component to the overall performance in the early fusion GEFA model, we remove each component from the GEFA model. We conduct the ablation experiment using the Davis dataset benchmark in the warm setting. First, we evaluate the usage of the embedding feature by comparing it with the one-hot encoding. Second, we evaluate the usage of attention mask as the graph edge. Instead of using attention as drug-residue edge weight, drug-residue edges are weighted the same as the residue-residue edges in the target graph. Third, we evaluate the usage of the 2-layer GCN and the usage of residual blocks to refine graph structure. As all residual blocks have shared weight, this reduces the number of parameters, which may help in the case of over-fitting. Fourth, we evaluate the residual blocks. We test three cases: without residual blocks in both protein and drug graph, without protein graph residual blocks, and without drug graph residual blocks. Finally, we compare the drug representation extracted from the drug-protein fusion graph and drug representation extracted from the drug graph. Instead of fusing two types of drug features followed by pooling, we only use one type of drug feature to combine with graph-based protein representation.

As shown in Table. 2, model using the protein embedding feature has an improvement of 11.85% in MSE and 1.00% in CI. This emphasizes the advantage of using the protein embedding feature as the graph node feature.

Table 2: The result of experiment warm setting with different component in Davis dataset in warm setting. The first row shows proposed GEFA with all components.)

Architecture	RMSE↓	MSE↓	Pearson↑	Spearman↑	CI ↑
GEFA	**0.4775**	**0.2280**	**0.8467**	0.7023	0.8927
One-hot encoding	0.5050	0.2551	0.8274	0.6919	0.8837
W/o attention	0.4887	0.2388	0.8392	0.7014	0.8909
W/o 2-layer GCN	0.4844	0.2346	0.8425	0.6933	0.887
Residual blocks usage					
W/o residual blocks	0.4933	0.2434	0.8351	0.686	0.8819
Drug graph res. blocks	0.4944	0.2444	0.8351	0.6874	0.8828
Protein graph res. blocks	0.4873	0.2375	0.8407	0.7045	0.8941
Drug representation choice					
Before fusion rep.	0.4806	0.231	0.8448	**0.7058**	**0.8954**
After fusion rep.	0.5171	0.2673	0.8174	0.6437	0.8558

Using attention mask as the drug-residue edge in protein graph gains improvement of 4.74% in MSE and 0.20% in CI. Without the attention mask, we assume that the drug molecule interacts with all residues in target protein equally. However, each residue contributes differently to the binding process. Therefore, it is reasonable to use self-attention to learn each residue's contribution level which is used as edge weight between drug node and residue node.

Using the 2-layers GCN (GEFA in Table. 2) shows improvement compared to without using 2-layer GCN (w/o using the 2-layers GCN in Table. 2). Compared to the model using solely residual blocks as graph refinement (w/o 2-layer GCN in Table. 2), the model with residual block shows an advantage over the model with only the 2-layers GCN (w/o graph residual blocks in Table. 2). Therefore, it can be suggested that residual blocks have the same or even better learning ability than the 2-layer GCN. Interestingly, combining both the 2-layers GCN and residual blocks brings the best result as shown in the full component model GEFA.

The model using residual blocks in both drug and protein graph shows an advantage over the model without any residual blocks. Model having residual blocks for both drug graph and protein graph gains 6.73% improvement in MSE over the model without residual blocks. It is interesting that applying residual blocks only for drug graph slightly decrease model performance (0.2% decrease in MSE and 0.1% decrease in CI). Adding back residual blocks for protein graph helps the model to gain 2.43% improvement in MSE. Stacking residual blocks in protein graph is more crucial in the early fusion approach as it affects not only protein graph representation but also the drug node representation.

Finally, we compare the drug representation before and after the drug-target graph fusion. Model using only drug representation before fusion shows comparable performance while the model using drug representation after fusion suffers 17.24% performance loss in MSE. This indicates that drug representation before fusion is more useful than after fusion. The reason is likely due to message passing in graph neural network. The drug node info is updated from its neighbor residue nodes. Therefore, this suggests that the binding process does not bring any signification change to the ligand latent representation.

5 Conclusion

We have proposed a novel deep learning method, called GEFA (Graph Early Fusion for binding Affinity prediction) for target-drug affinity prediction, a crucial task for rapid virtual drug screening and drug repurposing. To improve the power of protein representation, we use self-supervised to take advantage of a large amount of unlabeled target sequences. To address the latent representation change due to conformation change during the binding process, the early fusion between drug and target is proposed. Unlike the late fusion approach extracting representation separately, the early fusion approach integrates drug representation info into protein representation learning phase. The self-attention value of the target sequence is used as edge weight connecting drug node and residue node in the target protein graph. Self-attention allows the model more interpretable as it shows which residues contribute to the binding process and the level of contribution of each residue. The quantitative experiments show that the early fusion approach has advantages over the late fusion approach. Using the embedding feature as target node feature has advantages over using one-hot encoding. Residual block design allows stacking multiple GCN layers for better learning representation capability.

References

References

[1] M. Thafar, A. B. Raies, S. Albaradei, M. Essack, and V. B. Bajic, "Comparison study of computational prediction tools for drug-target binding affinities," *Frontiers in Chemistry*, vol. 7, 2019.

[2] M. K. Gilson, T. Liu, M. Baitaluk, G. Nicola, L. Hwang, and J. Chong, "BindingDB in 2015: a public database for medicinal chemistry, computational chemistry and systems pharmacology," *Nucleic Acids Research*, vol. 44, no. D1, pp. D1045–D1053, 2016.

[3] A. Cichonska, B. Ravikumar, E. Parri, S. Timonen, T. Pahikkala, A. Airola, K. Wennerberg, J. Rousu, and T. Aittokallio, "Computational-experimental approach to drug-target interaction mapping: a case study on kinase inhibitors," *PLOS Computational Biology*, vol. 13, no. 8, e1005678, 2017.

[4] A. Cichonska, T. Pahikkala, S. Szedmak, H. Julkunen, A. Airola, M. Heinonen, T. Aittokallio, and J. Rousu, "Learning with multiple pairwise kernels for drug bioactivity prediction," *Bioinformatics*, vol. 34, no. 13, pp. i509–i518, 2018.

[5] H. Öztürk, A. Özgür, and E. Ozkirimli, "DeepDTA: deep drug–target binding affinity prediction," *Bioinformatics*, vol. 34, no. 17, pp. i821–i829, 2018.

[6] T. Nguyen, H. Le, and S. Venkatesh, "GraphDTA: prediction of drug–target binding affinity using graph convolutional networks," *bioRxiv*, p. 684 662, 2019.

[7] H. Öztürk, E. Ozkirimli, and A. Özgür, "WideDTA: prediction of drug-target binding affinity," *arXiv preprint arXiv:1902.04166*, 2019.

[8] S. Zheng, Y. Li, S. Chen, J. Xu, and Y. Yang, "Predicting drug–protein interaction using quasi-visual question answering system," *Nature Machine Intelligence*, vol. 2, no. 2, pp. 134–140, 2020.

[9] S. J. Teague, "Implications of protein flexibility for drug discovery," *Nature Reviews Drug discovery*, vol. 2, no. 7, pp. 527–541, 2003.

[10] M. Jiang, Z. Li, S. Zhang, S. Wang, X. Wang, Q. Yuan, and Z. Wei, "Drug–target affinity prediction using graph neural network and contact maps," *RSC Advances*, vol. 10, no. 35, pp. 20 701–20 712, 2020.

[11] M. I. Davis, J. P. Hunt, S. Herrgard, P. Ciceri, L. M. Wodicka, G. Pallares, M. Hocker, D. K. Treiber, and P. P. Zarrinkar, "Comprehensive analysis of kinase inhibitor selectivity," *Nature Biotechnology*, vol. 29, no. 11, pp. 1046–1051, 2011.

[12] W. Ma, L. Yang, and L. He, "Overview of the detection methods for equilibrium dissociation constant KD of drug-receptor interaction," *Journal of Pharmaceutical Analysis*, vol. 8, no. 3, pp. 147–152, 2018.

[13] W. Torng and R. B. Altman, "Graph convolutional neural networks for predicting drug-target interactions," *Journal of Chemical Information and Modeling*, vol. 59, no. 10, pp. 4131–4149, 2019.

[14] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 4171–4186.

[15] R. Rao, N. Bhattacharya, N. Thomas, Y. Duan, P. Chen, J. Canny, P. Abbeel, and Y. Song, "Evaluating protein transfer learning with TAPE," in *Advances in Neural Information Processing Systems*, 2019, pp. 9689–9701.

[16] T. N. Kipf and M. Welling, "Semi-Supervised Classification with Graph Convolutional Networks," in *Proceedings of 5th International Conference on Learning Representations, ICLR 2017.*

[17] A. F. Agarap, “Deep Learning using Rectified Linear Units (ReLU),” *arXiv preprint arXiv:1803.08375*, 2018.

[18] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[19] T. Minh Le, V. Le, S. Venkatesh, and T. Tran, “Dynamic language binding in relational visual reasoning,” in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pp. 818–824.

[20] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[21] R. Rao, N. Bhattacharya, N. Thomas, Y. Duan, X. Chen, J. Canny, P. Abbeel, and Y. S. Song, “Evaluating protein transfer learning with TAPE,” in *Advances in Neural Information Processing Systems*, 2019.

[22] S. Wang, S. Sun, Z. Li, R. Zhang, and J. Xu, “Accurate de novo prediction of protein contact map by ultra-deep learning model,” *PLOS Computational Biology*, vol. 13, no. 1, e1005324, 2017.

[23] M. Fey and J. E. Lenssen, “Fast Graph Representation Learning with PyTorch Geometric,” in *Proceedings of ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019.

[24] M. Gönen and G. Heller, “Concordance probability and discriminatory power in proportional hazards regression,” *Biometrika*, vol. 92, no. 4, pp. 965–970, 2005.

[25] J. Benesty, J. Chen, Y. Huang, and I. Cohen, *Noise reduction in speech processing*. Springer Science & Business Media, 2009, vol. 2.

[26] D. Zwillinger and S. Kokoska, *CRC standard probability and statistics tables and formulae*. CRC Press, 1999.