Cross Attention DTI: Drug-Target Interaction Prediction with Cross Attention module in the Blind Evaluation Setup

Kyohei Koyama
SyntheticGestalt Ltd
London, UK
k.koyama@syntheticgestalt.com

Kotaro Kamiya
SyntheticGestalt Ltd
London, UK
k.kamiya@syntheticgestalt.com

Koki Shimada
SyntheticGestalt Ltd
London, UK
k.shimada@syntheticgestalt.com

ABSTRACT

We achieve the state-of-the-art performance for a drug-target interaction (DTI) prediction task with a Transformer-based neural network model. By serializing SMILES, fingerprints and protein sequence data for pairs of compounds and proteins we achieved promising prediction of DTI. The model improves the mean squared error metric compared to state-of-the-art models on two benchmark datasets. By using two Transformer encoders as feature extractors and the Cross Attention as a task executor, key regions of interest for novel drug candidates were found, allowing for structural highlighting of the compounds. Notably, compounds and proteins are referenced to each other to attend mutually by the attention mechanism. Additionally, we propose a model evaluation method that we call the blind evaluation, which is designed for the practical purpose of drug discovery. The model presented is conceived as a potential first screening method for mining large datasets of compounds and highlighting new potential drug candidates, as well as providing rich annotations to the structure of these compounds to inform High Throughput Screening (HTS) studies.

CCS CONCEPTS
- Computing methodologies → Neural networks; - Information systems → Information retrieval; - Applied computing → Bioinformatics.

KEYWORDS
Machine Learning, Deep Learning, Transformer, Drug-Target Interaction, Molecule Representation, Drug Discovery

1 INTRODUCTION

1.1 Background

The interactions between compounds and proteins guide and regulate an enormous variety of processes in the natural world. Within the cell, these interactions work to regulate protein behavior and (through downstream effects) the behavior of the cell, and organism as a whole [2].

The study of these interactions plays a vital role in the development of a broad class of drugs, where the activation, inhibition, or conformational change of a functional protein has some therapeutic benefit to a patient [1]. Development of these drugs is a long and expensive process. An average of around 5.5 years and 281 million US$ was spent per new drug last decade, with many large pharmaceutical companies now drawing back from drug development [20]. Prediction of protein-compound interaction which outperforms or augments traditional HTS methods has tremendous potential to accelerate the process of drug discovery, and bring the cost of development within reach of a wider community of researchers [24]. This study presents a state-of-art deep learning model for predicting Drug-Target Interaction (DTI), along with novel tools for highlighting regions of interest in the predicted compounds.

In general, DTI prediction models take a compound and a protein as inputs and predict a value associated with the relationship between them, such as a binding affinity. Researchers can use DTI models in two ways: in silico virtual screening and drug repositioning. Virtual screening can be used to identify any compounds that may interact with a given protein from a broad list of candidates. In drug repositioning, DTI models can be used with a drug (or drug candidate) to predict alternative drug targets and side effects (interactions with off-target proteins). The virtual screening system provides a means for conducting early-stage prioritization of drug candidates from the vast number of compounds in compound libraries.

The model presented here takes the primary sequence of a protein of interest and the SMILES representation of a given compound and categorizes them as interacting or non-interacting. The model uses a modern deep learning algorithm to perform this categorization. In addition, by taking the Morgan fingerprint (or ECFP) [22] of each molecule, the SMILES strings are decorated to differentiate different chiralities of the same compound, an extremely important factor in the development of new drugs and a novel method which is presented in detail in this study. We achieve promising candidate screening results for a variety of target proteins. Our model’s mean squared error (MSE) outperforms one of the state-of-the-art models, DeepDTA [19], on two datasets by 6.5% and by 9.8% and achieves a 1.3% C-index improvement on one dataset.
1.2 Task Definition
This study addresses two tasks: a regression task and a binary prediction task. The model takes a compound-protein pair as input to output a value.

- The regression task aims to predict a continuous value. The output values are real values that differ for each dataset, while they all express interaction strength between compounds and proteins.
- The binary classification task aims to predict whether an interaction is positive or not, meaning “does interact” or “does not interact”.

For the binary classification task, the target binary values of the model are created using experimental bioactivity data, such as the inhibition constant $K_i$, the dissociation constant $K_d$, and the “inhibitory concentration 50%”, IC50 which measure the capacity of the compound to interact with a target protein in various ways. These values are determined experimentally and cataloged for protein-compound pairs in various datasets.

Binarizing different datasets of real values with some threshold is essential to uniformly deal with different experimental values and to make use of more substantial data for industrial use. However, this is a challenging task, as biological data has no de facto standard across various datasets. (see Section 4.2 for details).

1.3 Challenges
The development of any DTI model must contend with the following four challenges.

**Application to unseen data:** The problem of prediction in a DTI model is similar to the cold start problem in recommendation systems [34], where a system recommends items to the user which the systems (and user) have not seen before. Likewise, a DTI model should be able to predict an interaction for unseen proteins or compounds.

To create the “unseen” situation, paired data offers the additional challenge that data cannot be randomly removed naively. This implies that we cannot remove an entire class of proteins or an entire species of compounds from the training dataset. Keeping one-half of the paired data in the training data can affect the model performance, and creates the possibility of false positives.

**Careful dataset selection:** In some drug-target datasets, as a result of dataset bias, target protein information is not required to predict the interactions [4]. Notably, datasets with artificially created samples, such as DUD-E [17] and Liu [14] used in Tsubaki et al. [28], make it easy to distinguish a positive interaction from a negative interaction by observing only compounds.

**Careful representation of compounds:** The small number of interactions per compound makes the prediction become impractically easy, by connecting the specific feature of compounds to the interaction prediction. Accordingly, we should avoid using a strong feature representation that distinguishes a compound from others since the model get influenced only by the specific feature [4].

This means that when a feature consists of vocabulary with a large subgraph, the feature can become so specific that several substructures of the graph should be avoided in use.

**Enabling interpretation of prediction:** When a DTI model is used for practical drug design, the interpretability of the neural network model is essential. Observing positions where proteins and compounds bind to each other is not only helpful to understand the model prediction but also critical to run docking simulations to identify the protein binding pocket.

1.4 Contributions
This study makes four major contributions.

- **Application to unseen data:** We have conducted an experiment where the evaluation is made without any compounds and proteins used in training. Since the model newly encounters both compounds and proteins here, it is beneficial in practical use such as virtual screening and drug repositioning. Section 4.2 describes the details of creating datasets.

- **Careful dataset selection:** We have calculated the ratio of how many unique compounds or proteins have both positive and negative interactions in several datasets, as Table 1 shows. It suggests that the interactions in DUD-E and Liu can easily be predicted from only compounds without considering the pair of compound and protein.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Compounds</th>
<th>Proteins</th>
</tr>
</thead>
<tbody>
<tr>
<td>Liu</td>
<td>0.0090</td>
<td>0.1268</td>
</tr>
<tr>
<td>DUD-E</td>
<td>0.0000</td>
<td>1.0000</td>
</tr>
<tr>
<td>KIBA</td>
<td>0.7902</td>
<td>0.9913</td>
</tr>
<tr>
<td>Davis</td>
<td>1.0000</td>
<td>1.0000</td>
</tr>
<tr>
<td>Blind Eval. Dataset</td>
<td>0.3105</td>
<td>0.1463</td>
</tr>
</tbody>
</table>

**Careful representation of compounds:** We improve the natural language processing (NLP) based compound representation from the existing study [11] and apply it to this model. The representation can capture the chirality and topological information extracted from canonical SMILES (Section 3.2). We evaluate the model with different feature representation to find out the best representation to avoid overly specific tokens for compounds by controlling the size of vocabularies, which exponentially increases with an increase in the radius size of substructure around each atom.

**Enabling interpretation of prediction:** By looking into the attention weights in the attention layer and highlighting regions which are given particular focus, as described in Vaswani et al. [29], we can interpret the interaction mechanism of DTI (Section 3.3). In this study, the multi-head attention network from Transformer [29] is modified for compounds and proteins to attend each other mutually. Finally, we get the attention weights to see which part of proteins attend to which part of compounds and vice versa.

2 RELATED WORK

2.1 Virtual Screening
We must pay ample attention when evaluating machine learning (ML) based models against non-machine-learning models that include docking simulation, similarity search, and 3D-based modeling.
[3, 18, 21]. Notably, Zhu et al. [38] stated that prospective studies have lower hit-rate (precision) than retrospective studies. Prospective studies are those aiming to discover new compounds that are not in a dataset, while retrospective studies are those that compare existing data for benchmarking. They argued that the precision of prospective virtual screening is considered outstanding when over 13% for the prospective studies. The ML-based DTI model is a type of those virtual screening methods that often clearly presents better scores than the existing non-ML virtual screening models [28, 31] since it measures the performance using the retrospective setting. However, we must be more careful regarding the evaluation method and the dataset design, which is why we propose a new evaluation setting in this study.

2.2 Machine Learning Based DTI Prediction

The advantage of ML models over others is that ML does not require specific data properties. ML-based DTI models predict interactions of a given protein-compound pair and these predictions take any features of compounds and proteins as input. Through the use of a neural network and an extensive training, they output the final value. DeepDTA [19] and a study by Tsubaki et al. [28] are examples of such studies.

DeepDTA implemented DTI with CNN on labels of encoded SMILES and with CNN on labels of encoded amino acid sequences. As such, their model is a baseline model to compare in this study. They took two datasets and reported MSEs of 0.194 on the KIBA value. DeepDTA [19] and a study by Tsubaki et al. [28] is more suitable here for protein sequence. The functionality of the protein sequence is dependent on several amino acid blocks [29] since it measures the performance using the retrospective setting. Therefore, we use the compound feature design method similar to Mol2Vec [11] as an extension of ECFP. Unlike Mol2Vec, we do not use a vector representation of compounds. Instead, we analogize the NLP to represent compounds by using Transformer, which has been producing great results in NLP because we consider capturing the global information of atom sequences is necessary for DTI.

On the other hand, proteins are composed of 20 types of amino acid sequences (e.g. MENATL...LKST). A protein at its most basic level is a folded polymer chain of various species of amino acids. This poly-peptide chain will fold during synthesis to form a three-dimensional protein. This becomes this final shape (or conformation) along with the primary amino acid sequence that determines the behavior of the protein.

A method called structure-based drug discovery assesses the 3D structures of proteins and attempts to identify drugs that bind against proteins using their 3D coordinates. Docking simulation is used for this purpose and sometimes requires the crystal structure of the protein complex, the protein-bound with the compound. However, there are several problems with this technique. For example, it sometimes takes several months to obtain the 3D structure of proteins or inaccurate coordinates experimental instrument error might occur. In this case, researchers attempt to predict the 3D structure using ML [25] or by attempting to estimate the DTI using another method such as cell phenotype assay, which is very costly and time-consuming. Therefore, the DTI prediction based only on sequence information is required, and our proposed method is using only the protein sequence. We represent the primary sequence using a string of characters which represent amino acids with 26 alphabets.

3 METHODOLOGY

3.1 Notation

To begin, we illustrate notation, for short. We define a natural number $n \in \mathbb{N}$, $[n] = \{1, \ldots, n\}$ and $(x_i)_{i \in [n]} = (x_1, \ldots, x_n)$.

3.2 Representation

Compounds: Formally, the procedure of creating a compound feature is described as a map from $G \mapsto G'$. The source is a compound: a graph $G = (A, B)$ which has labeled nodes and labeled edges. Let $N$ be the maximum number of atoms, $A$ be a set of all atom tokens (letters) and $B$ be a set of all bond tokens. For a technical reason, the atom tokens and the bond tokens include the Null token to express non-existing. In the source compound,
\[ A = \{a_i | a_i \in A, i \in [N]\}, \]
\[ B = \{b_{ij} | b_{ij} \in B, (i, j) \in [N] \times [N]\} \]
are sets of atoms and bonds. Next, the destination is a compound feature: a graph \( G' = (F, B) \), where \( F \) is a set of fragment tokens, and also considered a sequence.

\[ F = \{f_i | f_i \in F, i \in [N]\}. \]

Here, \( F \) is a set of all fragment tokens which are specified with a given fingerprint algorithm. Therefore, it is essentially a relabeling function \( f_i = t \) which finds a corresponding fragment token with a retrieval query: a graph generated \( Q_i \in G_A \) from \( a_i \) with a given maximum radius \( R \). Figure 2 illustrates the steps above. Finally, we use \( F \), ejected from \( G' \), as a compound feature since each fragment token includes local connection information inside and our model can capture global relationships between tokens.

![Figure 1: Example of the canonicalization which changes the indices of atoms. The left is the SMILES before canonicalization. The right is the one after canonicalization. The yellow shades indicate reordered atoms.](image)

**Proteins:** Only sequence information is used in our model: One data record of protein is a sequence of amino acids. Similarly, in conventional natural language processing, several amino acids are working together, which are known as tripeptides. Tripeptides are short chains of amino acids linked by peptide bonds. It is known that tripeptides are more useful in predicting protein functionality than a single amino acid. As such, we have encoded all the sequences of target proteins into a sequence of tripeptides. A tripeptide is constructed such that one amino residue contains both adjacent amino residues, starting at the second residue and terminating at the second from the end. Eventually, the tripeptide sequence is two shorter than the original. Since the number of types of amino acids is at most 26 including special types, the maximum vocabulary size of tripeptides is only \( 26^3 = 17,576 \) and it does not cause the dataset bias.

### 3.3 Model

Our model architecture is represented in Figure 3. When handling paired data in particular, we propose the use of building blocks for integrating two different information types, which are the Cross Transformer Encoder in Figure 4 and, more importantly, the Cross Attention module in Figure 5. The model feeds a left input (compound feature) and a right input (protein feature).

![Figure 2: There are two steps to get \( f_i \) from \( a_i \). Step 1: Generating a query fragment \( Q_i \) from \( a_i \) by taking the largest radius of no more than \( R \). Step 2: Finding the corresponding fragment token \( f_i \) and set \( f_i = t \) until filling out \( F \).](image)

First, each input is converted by the Input Embedding (summing up token embedding and positional encoding) [6, 29] and the Transformer Encoder [29] following the original methods. Then, the two resultants \((u, v)\) are further encoded to \((u', v')\) by the Cross Transformer Encoder while retaining their shapes. At the top of the model, the output is a real value for a regression task or class probabilities for a classification task. The detail of the parameters is presented in the Appendix.

The Cross Attention module is the heart of the Cross Transformer Encoder as figures show. We derive the Cross Attention module from a general definition by extending the generic non-local operation [32] which includes a wide class of operations such as self-attention. The extended version of the generic non-local operation is reformed to be explicitly applicable to paired data. We call it a referencing map and define it as described below.

**Definition 1 (Referencing Map).** Let \( U, V \) be finite-dimensional vector spaces over \( \mathbb{R} \) and \( n, m \in \mathbb{N} \). For instance, assume \( V = U = \mathbb{R}^d \) where \( d \) is embedding dimension and \( n, m \) are size or length of each input. Here, a referencing map \( r : U^n \times V^m \rightarrow U^n \) is defined as follows. For \((u, v) \in U^n \times V^m\)

\[ r(u, v) = \left( r^i(u, v) \right)_{i \in [n]} \]

and, for each \( i \in [n] \), the \( r^i : U^n \times V^m \rightarrow U \) is calculated as
Cross Attention DTI

\[ r^j(u, v) = \xi^j \left( \frac{1}{c_i(u, v)} \sum_{j=1}^{m} f^j_i(u, v) g^j(v) \right). \]

Here, we have four types of functions when \( i \) is fixed and for each \( j \in [m] \):

- \( \xi^j : W \to U \)
- \( c^j : U^n \times V^m \to \mathbb{R} \)
- \( f^j_i : U^n \times V^m \to \mathbb{R} \)
- \( g^j : V^m \to W \)

where \( \xi^j \) is usually a linear operator or an identity, \( c^j \) is an arbitrary normalization factor, \( f^j_i \) is a pairwise function to refer to opponent’s object, \( g^j \) is for feature transformation and \( W \) is another vector space.

**Definition 2 (Cross Referencing Map).** Let \( r_{\text{left}} \) and \( r_{\text{right}} \) be referencing maps on \( U^n \times V^m \) and \( V^m \times U^n \), respectively. We define cross-referencing: \( U^n \times V^m \to U^n \times V^m \) as follows: For \((u, v) \in U^n \times V^m\)

\[ \text{cross-referencing}(u, v) = \left( r_{\text{left}}(u, v), r_{\text{right}}(v, u) \right). \]

**Definition 3 (Cross Attention).** A cross referencing map is said to be a cross-attention, if

\[(U, V, W) = (\mathbb{R}^{d_u}, \mathbb{R}^{d_v}, \mathbb{R}^{d_w})\]

and the four types of functions are defined, with feature maps \( \phi^i : U \to W \), \( \psi^j : V \to W \) and \( \psi^j : V \to W \) for \((i, j) \in [n] \times [m] \), as

- \( \xi^j(w) = \text{linear}(w) \)
- \( c^j(u, v) = \sum_{j=1}^{m} f^j_i(u, v) \)
- \( f^j_i(u, v) = \exp \left( \phi^i(u') \cdot \psi^j(v') \right) \)
- \( g^j(v) = \psi^j(v) \)

where \( \cdot \) is an inner product. In addition,

\[ a^j_i = \frac{1}{c^j(u, v)} f^j_i(u, v) = \frac{f^j_i(u, v)}{\sum_{j=1}^{m} f^j_i(u, v)} \]

is called a cross-attention weights since it is normalized. Namely,

\[ \sum_{j=1}^{m} a^j_i = 1 \text{ holds for all } i \in [n]. \]

### 3.4 Training

We use the RAdam optimizer [15]. In terms of losses, we use MSE loss for the regression and the weighted cross entropy (CE) loss for the binary classification since the positive-negative ratio is imbalanced.

![Figure 3: The whole architecture of our model. Note that the ‘Mean’ is reducing the length dimension. The ‘Linear’ has one linear layer and a relu activation function and final linear layer that produces one value. The number of heads of each Transformer encoder was 4.](image)

![Figure 4: The Cross Transformer Encoder module.](image)

### 4 EXPERIMENT

#### 4.1 Experiment Setting

We conducted the following three experiments to demonstrate that 1) our model is the state-of-the-art, 2) it is practically useful for DTI purposes; and 3) the Cross Attention module is essential for the prediction. All the training were done with 5-fold cross validation.

**4.1.1 Comparison Test.** We test our model performance by comparing it to DeepDTA [19], which is one of the state-of-the-art studies for DTI purposes, as well as SimBoost [10]. The following briefly describes these models. Since DeepDTA is a regression model using continuous experimental bioactivity data, we also formed our model as a regression model.

![Diagram of DeepDTA](image)
where neither of them is included in the training dataset. Since the Attention module to evaluate how the vocabulary size affects to Table 2: The number of unique tokens for compound feature Transformer encoders and passes it to three-layered fully-connected module with the fully connected layer.
two 512 hidden nodes. Thus, we can compare the Cross Attention layer instead of using the Cross Attention module. The layers have model without the module concatenates the outputs from the two out the Cross Attention module to the binary classification task. The Cross Attention module, we have applied our model with and with-

4.1.3 Cross Attention Evaluation.

To evaluate whether our model can show practically high performance even for protein and compound pairs where neither of them is included in the training dataset. Since the proteins and compounds accessible as a training dataset are quite limited compared to the potential candidate pool for the virtual screening and the drug repositioning purpose, the model should be able to accurately predict interactions for unseen data. We also test the different size of radius of compound features with the Cross Attention module to evaluate how the vocabulary size affects to the module. Table 2 shows such a vocabulary size by the radius of compound feature.

Table 2: The number of unique tokens for compound feature in the datasets used in this study: Davis, KIBA, and the blind evaluation dataset after the fingerprint algorithm is applied.

<table>
<thead>
<tr>
<th>Radius</th>
<th>The number of unique tokens for compound feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>R=1</td>
<td>969</td>
</tr>
<tr>
<td>R=2</td>
<td>16,983</td>
</tr>
<tr>
<td>R=3</td>
<td>54,227</td>
</tr>
</tbody>
</table>

4.2 Datasets

In all experiments, we used only compounds which consist of not more than 150 atoms and proteins whose sequence length is not more than 2,500 since attention layers consume computational memory proportional to the square of the length.

4.2.1 Comparison Test. We use two datasets, KIBA and Davis, which are used in DeepDTA with the exact same train and test split according to their GitHub repository 1. The KIBA dataset has 2,111 unique compounds and 229 unique proteins, while the Davis dataset has 68 unique compounds and 442 unique proteins. Table 3 presents the statistics of the dataset. They divide the dataset equally into six subsets and take one subset as a test dataset, then use the others for training with five-fold cross-validation. We followed the same steps

4.2.2 Blind Evaluation and Cross Attention Evaluation. A blind evaluation was performed to evaluate our model with proteins and compounds that are not used for training. Therefore, we had to carefully choose interactions from the dataset as test data and discard those whose either protein or compound was used for training (see Figure 6). The Comparison Test shown above in 4.2.1 used the typical test dataset design illustrated in the left of Figure 6.

To demonstrate the general applicability of our model, we have used four different datasets KIBA [27], Metz [16], Davis [5], which are chosen based on Feng et al. [8], and DrugBank [36]. To unify these datasets with different formats as a single dataset, we bina-

rized the output value. The binarization method followed Tang et al. [27]. Here is an additional remark; we have avoided using artificial negative samples since it can induce false-positive predictions. The detailed dataset creation process is as follows,

- Step 1: Binarizing. We extracted pairs of canonical SMILES and target protein sequences and their affinity values from the datasets. For the KIBA dataset, the threshold was 12.1 to binarize the affinity (negative interactions are the ones with values less than 12.1). For the Davis dataset, the threshold was 7.0, while negative interactions were those with values less than 7.0. For the Metz dataset, the threshold was 7.6, while negative interactions were those with values less than 7.6. DrugBank dataset is already a binary dataset.

1 https://github.com/hklmzt/DeepDTA
All the experiments are conducted on a machine with an NVIDIA Tesla V100 GPU (16 GB GPU memory), 20-core Intel Xeon E5-2686 v4 CPU (2.20 GHz), and 244 GB of RAM. One fold of cross validation training took 24 hours for each dataset.

4.3 Metrics

4.3.1 Comparison Test. Regarding the regression problem where the numerical value of bioactivity of DTI is predicted, MSE and C-index were used as per DeepDTA. We use mean squared error (MSE) and the concordance index (C-index) [9]. C-index is a value according to the prediction, while 0.5 is expected for random predictions.

4.3.2 Blind Evaluation and Cross Attention Evaluation. We use PR-AUC and ROC-AUC. Since it is more important for decision-makers to avoid as in Section 1.3, the Cross Attention module performs better than the second-best by 2.055% and 0.6024 %, respectively. The second-best model as for PR-AUC is the model without the Cross Attention with R = 3 at 0.6326, followed by But the second-best as for ROC-AUC is the model without the Cross Attention with R = 3 at 0.7802.

5 RESULTS

5.1 Experiment Results

5.1.1 Comparison Test. Our model with the embedding of the canonical sequence fingerprint at radius 3 shows the best MSE among the other methods in the apples-to-apples comparison (Table 4) with DeepDTA and SimBoost [10]. The result improves DeepDTA’s score by 6.5% for Davis dataset and by 9.8% for KIBA dataset. Our model is able to improve the score at KIBA more than at Davis. Although, Table 4 shows our model improves 1.3 % C-index at Kiba, but DeepDTA (1) is better at Davis by 1.1 %.

5.1.2 Blind Evaluation. As for the binary classification model, we conduct experiments with different radial compound feature datasets and with the different models. In the R = 2 feature, the model (a) with the Cross Attention module is better and it showed the best metrics among all the datasets and models, as shown in Table 5, showing 0.6456 PR-AUC and 0.7849 ROC-AUC. These figures are better than the second-best by 2.055% and 0.6024 %, respectively. The second-best model as for PR-AUC is the model without the Cross Attention with R = 3 at 0.6326, followed by But the second-best as for ROC-AUC is the model without the Cross Attention with R = 3 at 0.7802.

5.1.3 Cross Attention Evaluation. Table 5 compares the model between with and without the Cross Attention module. When the radius is small, R = 1 and R = 2, the Cross Attention module contributes more than simple fully-connected layers. This means when the vocabulary does not include unique tokens, which is important to avoid as in Section 1.3, the Cross Attention module performs better. At R = 2, the Cross Attention model shows 2.3% higher PR-AUC and 0.6% higher ROC-AUC than the three fully-connected layer model. However, the score of Cross Attention with R = 3 is worst at this experiment, presenting 0.5989 PR-AUC.

Table 3: Dataset Statistics

<table>
<thead>
<tr>
<th></th>
<th>KIBA</th>
<th>Davis</th>
<th>Blind Evaluation (Test)</th>
<th>Blind Evaluation (Training)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unique compound count</td>
<td>2,111</td>
<td>68</td>
<td>2,580</td>
<td>6,278</td>
</tr>
<tr>
<td>Unique protein count</td>
<td>229</td>
<td>442</td>
<td>1,004</td>
<td>2,821</td>
</tr>
<tr>
<td>The number of interactions</td>
<td>118,254</td>
<td>30,056</td>
<td>28,912</td>
<td>104,669</td>
</tr>
<tr>
<td>Mean number of interactions per one compound</td>
<td>56</td>
<td>442</td>
<td>11</td>
<td>17</td>
</tr>
<tr>
<td>Mean number of interactions per one protein</td>
<td>516</td>
<td>68</td>
<td>29</td>
<td>37</td>
</tr>
<tr>
<td>Median number of interactions per one compound</td>
<td>39</td>
<td>442</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Median number of interactions per one protein</td>
<td>567</td>
<td>68</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 4: MSE on Davis and KIBA (Mean and Std). DeepDTA and Simboost rows are from [19]. Parentheses indicate the standard deviation

<table>
<thead>
<tr>
<th>Metric</th>
<th>Model</th>
<th>Davis</th>
<th>KIBA</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSE</td>
<td>Ours (R=3)</td>
<td>0.244 (0.004)</td>
<td>0.175 (0.003)</td>
</tr>
<tr>
<td></td>
<td>DeepDTA (1)</td>
<td>0.261</td>
<td>0.194</td>
</tr>
<tr>
<td></td>
<td>DeepDTA (2)</td>
<td>0.420</td>
<td>0.204</td>
</tr>
<tr>
<td></td>
<td>SimBoost</td>
<td>0.282</td>
<td>0.222</td>
</tr>
<tr>
<td>C-index</td>
<td>Ours (R=3)</td>
<td>0.876 (0.066)</td>
<td>0.874 (0.001)</td>
</tr>
<tr>
<td></td>
<td>DeepDTA (1)</td>
<td>0.878 (0.004)</td>
<td>0.863 (0.002)</td>
</tr>
<tr>
<td></td>
<td>DeepDTA (2)</td>
<td>0.886 (0.008)</td>
<td>0.854 (0.001)</td>
</tr>
<tr>
<td></td>
<td>SimBoost</td>
<td>0.872 (0.002)</td>
<td>0.836 (0.001)</td>
</tr>
</tbody>
</table>
Table 5: Metrics on PR-AUC and ROC-AUC. The model (a) indicates the Cross Attention model and (b) uses the fully-connect layers instead.

<table>
<thead>
<tr>
<th>Metric</th>
<th>Model</th>
<th>R=1</th>
<th>R=2</th>
<th>R=3</th>
</tr>
</thead>
<tbody>
<tr>
<td>PR-AUC</td>
<td>(a) 0.6247 (0.0046)</td>
<td>0.6456 (0.0090)</td>
<td>0.5989 (0.0453)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(b) 0.6211 (0.0118)</td>
<td>0.6312 (0.0140)</td>
<td>0.6326 (0.0124)</td>
<td></td>
</tr>
<tr>
<td>ROC-AUC</td>
<td>(a) 0.7712 (0.0034)</td>
<td>0.7849 (0.0027)</td>
<td>0.7453 (0.0495)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(b) 0.7708 (0.0079)</td>
<td>0.7802 (0.0060)</td>
<td>0.7790 (0.0073)</td>
<td></td>
</tr>
</tbody>
</table>

5.2 Interpretability

Figure 7 illustrates an example of the attention weights of the Cross Attention module. The ligand compound is SOTRASTAURIN (CHEMBL565612) and the target protein is Protein kinase C alpha, which is associated with chordoid glioma and papillary glioneuronal tumors.

We can confirm the attention layer captures the bonds between the compound and the protein. Figure 7 shows the atom number 0, C0 or the carbon atom, has the high attention value and it is binding to Phe350A. The amino acid at position 352 is also attended in the protein sequence, which includes position 351 and 353 because we used tripeptides. We can state our model architecture can interpret the chemical interaction and help to identify the binding site.

Figure 7: The complex of a compound and a protein that exists in KIBA dataset is presented. The mean of attention weights on each atom which is targeted by the protein in the Cross Attention module is illustrated as the color depth. Larger attention weights get darker the red gets: darker positions are targeted strongly by the target proteins. The canonical SMILES atom order is depicted next to each atom. The figure is created based on the binding information from RCSB PDB (rcsb.org) of PDB ID 3IW4 [30], as the real data of interaction. As illustrated in [30], black dashed lines indicate hydrogen bonds, salt bridges, and metal interactions. The green solid line shows hydrophobic interactions. Compared to the complex figure, we can clearly observe that the red positions with large attention values correspond to the positions with bonds and interactions, such as the atoms with index 0 and 28.

6 DISCUSSION

We have shown our Cross Attention DTI is competitive against DeepDTA and Simboost in Section 5.1.1 and the importance of Cross Attention module in Section 5.1.3. The physical relationship between a drug and a protein is captured by our Cross Attention module, as described in Section 5.2.

Comparison Test: We consider that this result is owed to our model architecture of Cross Attention and the NLP-based feature extraction by Transformers. Since KIBA has a lot more unique compounds than Davis, it shows greater improvement than Davis in terms of both C-index and MSE, meaning richer information on compounds such as the canonical sequence of fingerprint was captured by our model.

We changed the radius from 1 to 3 in the comparison test with only one fold training and determined that a radius of 3 is optimal since in the typical test evaluation setting specific vocabulary can distinguish the interaction. If one can obtain an enormous amount of experimental drug property data for a particular protein, this result suggests using a strong feature representation like $R = 3$.

Blind Evaluation: Especially, it should be emphasized that the best Cross Attention model achieves 0.6456 at PR-AUC, which can be considered outstanding according to the previous study [38]. We attribute the best score of radius 2 to the exponential vocabulary size increase described in Table 2 being prohibitively problematic, since a specific resultant vocabulary can exist only in one compound, having a strong bias and leading to over-fitting.

Cross Attention Evaluation: We have confirmed when the radius is small, $R = 1$ and $R = 2$, the Cross Attention module performs better, but the PR-AUC and ROC-AUC becomes significantly lower at $R = 3$. This would be because the specific vocabulary references the counterpart easily in the paired data in the Cross Attention, while the influence is limited in the fully-connected model due to the concatenation and the linear matrix product. Although our comparison was made with our test dataset and our models, this result indicates that when two sequential things interact with each other, the Cross Attention module can make a significant difference in the prediction. We can consider the further great opportunity of the Cross Attention on the interactions of protein-to-protein or any NLP tasks.

7 CONCLUSION

In this paper, we have presented a competitive Transformer-based model, namely Cross-Attention DTI, for drug-target interaction predictions. We leveraged the attention mechanism for paired data of sequence, together with a feature design method for compounds. This resulted in the significant advance of one of the state-of-the-art DTI models. Our experiment result shows that the Cross Attention DTI outperformed one of the state-of-the-art methods by 6.5% and by 9.8% in terms of MSE. The Cross Attention module also outperformed the fully-connected module by 2.3%.

We also proposed a new test method with the new dataset. The new large dataset called the blind evaluation dataset was developed to properly and blindly evaluate DTI prediction by removing compounds and proteins of the test dataset from the training dataset. This blind evaluation method was different from the standard evaluation manner of the train-test split, and it should help the drug data mining to predict the property of unknown drugs or proteins whose experimental data are not available.
We consider that the model can capture chirality and the global relationship of the compound, thanks to our compound features with Transformer. Within the experiment, we also found out the optimal compound feature, which can mitigate the dataset bias of paired data. Further future studies may employ the model with the Cross Attention. We consider the Cross Attention is effective and applicable in any sequential paired data tasks, such as protein-protein interaction prediction or NLP tasks.

REFERENCES


A  APPENDIX

A.1  Model Parameter

Table 6: Model parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>batch size</td>
<td>60</td>
</tr>
<tr>
<td>dropout on compound Transformer</td>
<td>0.1</td>
</tr>
<tr>
<td>dropout on protein Transformer</td>
<td>0.1</td>
</tr>
<tr>
<td>embed dim on compound Transformer</td>
<td>64</td>
</tr>
<tr>
<td>embed dim on protein Transformer</td>
<td>64</td>
</tr>
<tr>
<td>feed forward dim on compound Transformer</td>
<td>64</td>
</tr>
<tr>
<td>feed forward dim on protein Transformer</td>
<td>256</td>
</tr>
<tr>
<td>num loops on Cross Attention</td>
<td>1</td>
</tr>
<tr>
<td>num heads on Cross Attention</td>
<td>4</td>
</tr>
<tr>
<td>num heads on compound Transformer</td>
<td>4</td>
</tr>
<tr>
<td>num heads on protein Transformer</td>
<td>4</td>
</tr>
<tr>
<td>num on loops on compound Transformer</td>
<td>2</td>
</tr>
<tr>
<td>num on loops on protein Transformer</td>
<td>2</td>
</tr>
<tr>
<td>optimizer name</td>
<td>RAdam</td>
</tr>
<tr>
<td>optimizer beta1</td>
<td>0.9</td>
</tr>
<tr>
<td>optimizer beta2</td>
<td>0.999</td>
</tr>
<tr>
<td>optimizer eps</td>
<td>1e-08</td>
</tr>
<tr>
<td>optimizer lr</td>
<td>2e-04</td>
</tr>
<tr>
<td>optimizer weight decay</td>
<td>1e-05</td>
</tr>
</tbody>
</table>