Visual Explanations for Drug-Target Affinity Prediction using GNNs

Vishnu Vinod
Dept. of Computer Science & Engineering
IIT Madras
cs19b048@smail.iitm.ac.in

Ankita H. Murthy
Dept. of Electrical Engineering
IIT Madras
ee21b020@smail.iitm.ac.in

Course Project Report

CS6024: Algorithmic Approaches to Computational Biology

Contents

1 Introduction

Drug discovery and development constitute a complex and resource-intensive process aimed at identifying compounds that can effectively modulate biological targets to treat diseases while minimizing adverse effects. Central to this process is the elucidation of drug-target interactions (DTIs), which underpin the efficacy and safety of therapeutic interventions. The quantification of binding affinity between drugs and their respective targets provides critical insights into the strength and specificity of these interactions, often expressed through metrics such as dissociation constant (K_d), inhibition constant (K_i), or half-maximal inhibitory concentration (IC50). Regulatory approval for new drugs often extends up to 15 years from initial discovery. Identifying potential drug-target interactions (DTIs) and predicting drug-target affinity (DTA) accurately is pivotal in drug development and repurposing, given the multi-target nature of drugs and multi-genic basis of diseases. Recent progress in deep learning techniques along with the abundance of biomedical datasets has significantly enhanced the development of such techniques in understanding DTIs and DTAs. These advancements enable the integration of diverse data modalities including genomics, proteomics, and pharmacology.

Among deep learning architectures, graph neural networks (GNNs) have demonstrated remarkable efficacy in processing non-Euclidean spatial data [2], making them particularly well-suited for modeling molecular graphs. By capturing the intrinsic information encoded in molecular structures, GNNs enable accurate identification and prediction of drug-target interactions. Moreover, the attention mechanism has emerged as a potent tool for enhancing the interpretability of GNN-based predictions. In recent years, GNNs have been used leverage drug molecule structural data for DTI/DTA prediction with remarkable success [11, 20, 21]. Currently most GNN-based models used for DTI/DTA prediction which offer explanations for their predictions do so only for drug molecules. With recent advancements in GNN interpretability and explainability, as reviewed by Khan *et al.* [8], improving this aspect of current explainability methods for DTI/DTA prediction, to allow explanations for both the drug and target molecules, is an intriguing task, and, the primary focus of this research project. Its primary application would be to aid researchers in modifying drugs by identifying active regions in the target protein which could potentially interact with new drugs.

1.1 Related Work

DTI & DTA prediction using Graph Neural Networks

A number of deep learning models have been proposed for DTA prediction. Ozturk *et al.* [13] proposed DeepDTA, which is based on a character-based sequence representation approach and uses convolutional neural networks (CNNs) to learn features from the 1D representations of drugs and target proteins for DTA prediction. In order to overcome the loss of biologically relevant short subsequences by DeepDTA due to low signal-to-noise ratio and the use of full-length sequences, Ozturk *et al.* [14] further proposed WideDTA, which represents protein sequences and drug strings as a set of words rather than a set of characters.

In recent years, the inherent graph-like structure of molecules motivated the development of a number of GNN-based models for DTI and DTA prediction. Gao *et al.* [5] used long short term memory networks (LSTMs) and Graph Convolutional Networks (GCNs) and projected proteins and drug structures into dense vector spaces, predicting DTI. The use of GNNs to predict DTA, a continuous value, was first introduced by Nguyen *et al.* through GraphDTA [11], directly modeling drugs as molecular graphs while representing proteins as 1D sequences. To incorporate the spatial structure of proteins, Jiang *et al.* proposed DGraphDTA [7], extending GraphDTA by constructing protein graphs based on contact maps. Recently, DeepNC [20] proposed by Tran *et al.* using hypergraph convolutions [1] and MGraphDTA [21] proposed by Yang *et al.* using multi-scale GNNs and CNNs, have extended GraphDTA making remarkable improvements in performance.

Due to their high expressiveness and interpretable results, there is also a growing amount of work using attention-based GNN models for DTA and DTI prediction, as documented in [23]. However, the use of attention mechanisms introduces significant computational complexity [21].

Explainability of Graph Neural Network Predictions

With the advent of deep learning, neural networks have become near ubiquituous for a large variety of problems. However, unlike decision trees, neural networks do not offer easy interpretability or explainability of their outputs. This has brought into question the reliablity of neural networks as a decision making tool in critical applications like healthcare. Explainability, the focus of this work, refers to an ML model's capability of explaining itself by identifying subsections of the input which it relied on most heavily to make its decisions.

Saliency maps are a common method of visualizing the relative impact of various parts of the input on the predictions of an ML model. Several methods such as gradient-based saliency maps [19], Class Activation Mappings (CAMs) [24] and Excitation Backpropagation (EB) [22] were tailored specifically to solve this problem for ML models based on CNNs. In 2017, Selvaraju *et al.* [18] introduced Gradient-weighted CAM (GradCAM) which used the gradients “flowing" into the final convolutional layer of the network to generate saliency maps which highlight the most important regions of the image. The use of gradients in GradCAM removed the requirement of additional learning models in CAM, contributing to its robustness and flexibility as a method of generating explanations and spawning several variants which have now been extensively studied [3, 10, 12]

These methods were however limited by their applicability to grid-structured data, like in images, on which CNNs themselves were designed to be used. Pope *et al.* [16] extended this approach into non-Euclidean structured data by extending existing methods (GradCAM, EB etc.) to handle predictions made by GCNs described by Kipf & Welling [9]. In the context of DTI/DTA prediction, the most recent work by Yang *et al.* [21] introduced a novel explanainability method Grad-AAM.

1.2 Motivation

Most GNN-based models [11, 20, 21] for DTI/DTA prediction involve learning representations for the drug and protein molecules separately using GNN-based and CNN-based models respectively. The learnt representations are then concatenated and used for DTI/DTA prediction. Explainability of such methods has not been well studied in the literature, particularly for models which do not use an attention-based mechanism. In [21], Yang *et al.* propose Gradient-weighted Affinity Activation Mapping (GradAAM) - an attention free visual saliency map generation technique. Motivated by GradCAM based approaches [3, 10, 12, 18], which utilize the gradient of the output with respect to the activations flowing through the last convolutional layer of the model, GradAAM has been shown, albiet qualitatively, to effectively identify salient sub-structures in the drug molecule.

This paper also proposes a multi-scale GNN model for learning drug molecule representations (MGraphDTA) and is very effective in the DTI/DTA prediction task. The use of multi-scale features in MGraphDTA is central to the quality of explanations generated by GradAAM. A major drawback of this method however, is the lack of identification of active regions of the target protein sequence. In our work, we aim to extend the approach proposed in [21] to offer explanations which identify active regions in both the drug and the target, and quantify the extent of their interaction.

However, the problem of generating paired saliency maps for multiple input data, has not been well studied in computer vision literature. Paired explanations have previously been studied in the context of image similarity models by Plummer *et al.* [15]. The authors propose Salient Attributes for Network Explanation (SANE), which given a reference and an input image, offers paired explanations in the form of saliency map paired with an attribute that best explains the match. This method is novel in its utilization of a similarity score instead of classification labels to generate these explanations. However, the lack of well-defined attributes for drug-protein matching in the context of DTI prediction makes it infeasible for us to adapt the above approach to our setting. Instead, we use a novel metric in conjunction with a GradAAM-style [21] approach to generate saliency maps for the drug to understand the mechanics of DTI better.

1.3 Objectives

The overarching goal of this project is to develop a post-hoc gradient-based method for generating explanations for predictions made by GNN-based DTI/DTA prediction models, identifying active regions in the drug and corresponding interacting regions in the protein molecule.

Following a detailed literature review (detailed in Section 1.1), we aligned our work towards three sub-objectives, which are aimed at fixing specific flaws in the existing body of work as detailed in Section 1.2

- Algorithm for post-hoc generation of target saliency maps for GraphDTA type models.
- Quantitative analysis of both drug and target saliency maps using existing metrics [16, 18] in XAI (eXplainable Artificial Intelligence).
- Design of a novel metric to quantify relative conformity of drug-target saliency maps.

1.4 Contributions

Under each of the objectives set out above, our contributions were the following:

- **Generation of target saliency maps for GraphDTA type models**

  We implement an algorithm GradTAM, inspired by GradAAM, to generate saliency maps for the target protein. We successfully identify regions in the protein that contribute heavily to predictions made by a DTI prediction model. This is done by gradient weighting across channel activations. We also create a visualization of the generated target attention maps.

- **Quantitative analysis of both drug and target saliency maps**

  We analyze the generated explanations by calculating the fractional performance drops when the salient and non-salient features are occluded. We carry out these experiments for several thresholds as described in Section 3.2.

- **Designing novel metric to quantify relative conformity of drug-target saliency maps**

  We propose a new metric, dubbed **conformity**, which aims to quantify the co-reactivity of the salient regions identified via thresholding of drug and target saliency maps. We provide a justification for the design of the above metric and analyze the values it takes (and their implications) for various datasets.

These contributions are spread over sections 2.2, 2.3 and 3.3 with further details provided in the remaining sections of this project report.

2 Methodology

2.1 How does GradAAM work?

GradAAM [21] uses the gradient information flowing through the final convolutional block on the molecular graph embedding branch of GraphDTA type models. The gradients flowing through the last convolutional block are chosen since the graph convolutional layers retain the spatial information about the input, such as number of nodes in graph. Aggregation of the gradients of the output prediction with respect to each node feature, denoted by A_k^v, at this layer allows us to calculate the channel (feature) importance scores, denoted by α_k. This is done as follows:

$$\alpha_k = \frac{1}{|\mathcal{V}|} \sum_{v \in \mathcal{V}} \frac{\partial P}{\partial A_k^v} \tag{1}$$

The vertex set of the molecular graph is denoted by $\mathcal{V}$ and the output of the model is denoted by P. The fundamental idea of this approach is that the feature learnt by a channel is important to the prediction if any change in this feature causes a large change in the output. The channel importance scores capture the relative feature importances across multiple channels of the graph convolutional layer. We then perform a weighted combination of the channel importance scores and the forward activations across each channel as follows:

$$\tilde{\mathcal{S}}_D = \sum_k \alpha_k A_k = \frac{1}{|\mathcal{V}|} \sum_k A_k \sum_{v \in \mathcal{V}} \frac{\partial P}{\partial A_k^v} \tag{2}$$

This gives us the unnormalized drug saliency map (denoted by $\tilde{\mathcal{S}}_D$). We further use ReLU to smooth out negative activations to produce a sparser saliency map. We also use min-max normalization (denoted by $\mathcal{N}$) to clip the output values between 0 and 1. The final drug saliency map is given by:

$$\mathcal{S}_D = \mathcal{N}\left(ReLU\left(\tilde{\mathcal{S}}_D\right)\right) \tag{3}$$

While the original GradAAM algorithm was designed for, and works best with, multiscale concatenated features like in MGraphDTA, we can make use of it for other GraphDTA style methods too to identify salient regions in the drug molecule. We can also adapt this algorithm for application to the protein representation learning branch of GraphDTA style methods. Discussed in detail in Section 2.2, we will see that this allows us to generate saliency maps for the target protein sequence.

2.2 Gradient-weighted Target Activation Mapping (GradTAM)

Inspired by GradAAM proposed by Yang *et al.* in [21], we propose a method for generating saliency maps for the target protein sequence which can be applied to the protein encoding branch of all GraphDTA style methods. Our proposed method, dubbed GradTAM in a fashion similar to GradAAM, is a post-hoc, gradient-based algorithm and offers adaptability in application across various architectures.

There exists a fundamental structural distinction between the drug and target data. Thus the respective encoding branches of GraphDTA style models also utilize different neural network architectures to learn the representations. In order to learn robust representations for the molecular graph, various graph neural network layers are used such as graph convolutional layers, graph isomorphism layers and graph attention layers. Similarly for learning target representations, we make use of 1D convolutional layers.

Since convolutions preserve spatial information about the input, we are able to extend the approach proposed in GradAAM. First, the gradient of the output prediction, denoted by P, is calculated with respect to the activations passing through each channel of the final convolutional layer, denoted by A_k . Averaging these gradients across the input sequence yields a channel (feature) importance score, denoted by α_k, as follows:

$$\alpha_k = \frac{1}{|\mathcal{T}|} \sum_{x \in \mathcal{T}} \frac{\partial P}{\partial A_k^x} \tag{4}$$

where, $\mathcal{T}$ represents the input target protein sequence and x represents the constituent amino acids. The channel importance scores obtained thus are used to weight the activations passing through the respective channels. This yields the unnormalized saliency map, denoted by $\tilde{\mathcal{S}}_T$, as follows:

$$\tilde{\mathcal{S}}_T = \sum_k \alpha_k A_k = \frac{1}{|\mathcal{T}|} \sum_k A_k \sum_{x \in \mathcal{T}} \frac{\partial P}{\partial A_k^x} \tag{5}$$

Unlike graph convolutions, which naturally preserve the number of nodes in the graph, using 1D convolutional layers without padding (as is used in the original GraphDTA paper [11]) results in the saliency map $\tilde{\mathcal{S}}_T$ requiring resizing. The resize operation, denoted by $\mathcal{R}$ is carried out prior to the application of the ReLU activation and min-max normalization as follows:

$$\mathcal{S}_T = \mathcal{N}\left(ReLU\left(\mathcal{R}\left(\tilde{\mathcal{S}}_T\right)\right)\right) \tag{6}$$

It may be noted that the decrease in spatial dimensions of the saliency map compared to the input is a common feature of GradCAM-based approaches [3, 10, 12, 18] and simply resizing the saliency map using interpolation to fill in values as required, has been shown to be sufficient. As will be seen in Sections 3.3 and 3.5, explanations generated thus effectively capture important features in the input.

2.3 Quantifying Conformity of Drug and Target Explanations

Using the approaches discussed in previous sections, we can generate the saliency maps for both the drug and the target. Further, using thresholding, discussed in Section 3.2, we can split the input into salient and non-salient regions. We conduct experiments where we occlude (hide) the salient regions identified thus and note the fractional drop in the relevant performance metric over the test set. As will be discussed in Section 3.1, this is the fidelity of the explanation.

For a performance metric M, consider the performance of the model (unoccluded M_O) when occluding salient regions in the (i) drug (M_D) (ii) target (M_T) and (iii) in both drug and target (M_{DT}). Then the fractional performance drops, denoted by ΔP, can be calculated as:

$$\Delta P_i = \frac{M_i - M_O}{M_O} \quad \text{for} \quad i \in X = \{D, T, DT\} \tag{7}$$

We now conjecture that occlusion of the salient regions in the drug molecule, makes it unable to react with the target, and vice versa. Thus the performance drops observed when the drug or the target salient regions are occluded should be equal to each other and to the performance drop when both salient regions are occluded.

To quantify this sentiment mathematically, we propose to use the root mean square of the differences (scaled appropriately) between each pair of performance drops, yielding a new metric, dubbed **conformity of paired explanations** and denoted by ΔC. Mathematically we have:

$$\Delta C = 1 - \sqrt{\frac{(\Delta P_D - \Delta P_T)^2 + (\Delta P_{DT} - \Delta P_D)^2 + (\Delta P_T - \Delta P_{DT})^2}{2}} \tag{8}$$

For any performance metric M (C-index, R^2 score), ΔC is scaled to lie in the range 0 to 1, by multiplying the RMS term by $\sqrt{\frac{3}{2}}$. Furthermore we invert the scale such that higher values of conformity (ideally 1) correspond to well paired drug and target saliency maps. We show in Section 3.3 that this metric captures certain nuances of our problem setting effectively. However, significant work must still be done to analyze and propose more robust metrics to quantify the degree to which the drug and target saliency maps conform.

3 Experimental Work

For our empirical work, we use two datasets: Davis and KIBA, released as part of the Therapeutic Data Commons package [6]. The implementation of the various GNN based architectures was done using the open-source `pytorch` and `pytorch-geometric` [4] libraries. All experimentation was carried out on a local machine equipped with an NVIDIA RTX3050 GPU.

3.1 Performance Metrics

Fidelity and sparsity [16, 17] serve as two measures to gauge the quality of explanations.

Fidelity is defined [16] as a fractional drop in performance when the salient regions are occluded in the input. The salient regions can be identified by using a threshold in the activation maps generated by either GradAAM or GradTAM. Any “larger is better" performance metric may be used to measure this fractional drop. For our experiments we use the Concordance Index (CI or C-Index), which is the measure of the relative ordering of predictions compared to their ground truth values. CI is often used in conjunction with the MSE (Mean Squared Error) in DTI studies [11, 13, 14] as a performance metric with values ranging from 0 (reversed ordering) to 1 (perfect ordering).

Sparsity is a measure of the fraction of the input which is not selected as a salient region by an explanation [16, 17]. It may be defined in a similar fashion for both GradAAM and GradTAM. In case of a molecular graph saliency map with vertex set V, the sparsity is defined as $1 - \frac{|V_{salient}|}{|V|}$ where $V_{salient}$ denotes salient vertex set of the molecule. For the protein sequence saliency maps, the sparsity is given by $1 - \frac{\#salient\ amino\ acids}{\#sequence\ length}$. It may be noted that both these measures of sparsity involve the partition of the input sequence or graph into mutually exclusive exhaustive subsets of salient and non-salient inputs. This is normally done by selecting a threshold value for the activation mapping. As seen later in Section 3.2 the choice of thresholding is vital in analyzing the results.

3.2 Quantile Thresholding

In order to understand and evaluate the performance of our proposed method, we use thresholding to identify salient and non-salient input regions. Using both GradAAM and GradTAM on trained models which predict the output, we obtain drug and target saliency maps. However, for two saliency maps on inherently different data (molecular graph vs protein sequence), employing the same threshold to partition the inputs into salient and non-salient input regions is non-optimal for the analysis of the results due to the inherent difference in the distribution of values in either saliency map.

A straightforward workaround this obstacle would be to use separate thresholds for both the drug and target saliency maps. However, analyzing the explanations for multiple thresholds would lead to prohibitively large number of experimental settings ($2 \times 2 \times 3 \times 11 \times 11 \approx 1400$). This would make analysis of the results intractable.

In order to counter this issue, we propose to use a form of quantile thresholding to pick a fixed fraction of inputs which contributes most to the predictions ie. which has the highest values in the output saliency maps for both the drug and the target. This offers the added advantage that the explanations thus generated for both drug and target will have comparable sparsity. This makes the analysis of the results significantly more tractable compared to setting separate value thresholds for both drug and target saliency maps, with only $2 \times 2 \times 3 \times 11 \approx 130$ experimental settings in total across all models and datasets we consider.

We also report the results obtained when using a straightforward mean thresholding for both drug and target saliency maps in order to compare with the median thresholding ($50\%ile$ threshold) results.

3.3 Quantitative Results

We carry out experiments for 10 different values of quantile thresholds, along with mean thresholds, under three different occlusion settings: occluding only the drug, only the target and both the drug and the target. Experiments are conducted for GraphDTA models with 2 different drug encoding branches: GCN (Graph Convolutional Network) and GIN (Graph Isomorphism Network).

The results are presented below across two tables, one for each model.

With only salient regions				Occluding salient regions				
Thresh	$\Delta P'_D$	$\Delta P'_T$	$\Delta P'_{DT}$	Thresh	ΔP_D	ΔP_T	ΔP_{DT}	ΔC
10	0.101	0.087	0.113	90	0.232	0.081	0.245	0.801
20	0.129	0.111	0.157	80	0.301	0.121	0.345	0.794
30	0.152	0.152	0.221	70	0.337	0.155	0.358	0.807
40	0.203	0.221	0.338	60	0.354	0.184	0.316	0.845
50	0.264	0.307	0.456	50	0.379	0.215	0.311	0.857
60	0.313	0.374	0.499	40	0.399	0.242	0.319	0.864
70	0.363	0.404	0.489	30	0.425	0.262	0.337	0.858
80	0.386	0.408	0.429	20	0.435	0.272	0.334	0.857
90	0.398	0.400	0.357	10	0.412	0.309	0.373	0.910
μ	0.296	0.399	0.488	μ	0.369	0.249	0.345	0.890

Table 1: Fractional drop in performance (CI) with input occlusion for GCN model tested on Davis dataset; Thresholds given in percentile; μ denotes mean thresholding; ΔC is calculated using Eq 8

With only salient regions				Occluding salient regions				
Thresh	$\Delta P'_D$	$\Delta P'_T$	$\Delta P'_{DT}$	Thresh	ΔP_D	ΔP_T	ΔP_{DT}	ΔC
10	0.068	0.070	0.086	90	0.246	0.066	0.266	0.809
20	0.112	0.093	0.139	80	0.277	0.100	0.322	0.797
30	0.148	0.115	0.184	70	0.306	0.144	0.366	0.801
40	0.207	0.152	0.262	60	0.332	0.190	0.395	0.818
50	0.269	0.213	0.357	50	0.331	0.239	0.402	0.858
60	0.302	0.277	0.420	40	0.333	0.272	0.420	0.871
70	0.314	0.313	0.421	30	0.345	0.300	0.439	0.877
80	0.308	0.331	0.414	20	0.348	0.322	0.444	0.888
90	0.320	0.326	0.370	10	0.384	0.337	0.444	0.907
μ	0.302	0.277	0.419	μ	0.332	0.190	0.395	0.818

Table 2: Fractional drop in performance (CI) with input occlusion for GIN model tested on Davis dataset; Thresholds given in percentile; μ denotes mean thresholding; ΔC is calculated using Eq 8

Due to computational constraints, both time and hardware, we carry out the quantitative analysis only on the smaller Davis dataset. However, we visualize the drug and protein saliency maps for both Davis and KIBA datasets in Section 3.5. Further, we carry out experiments by occluding the salient regions, to show performance drop when these regions are absent, as well as by occluding non-salient regions, to show that features captured in the salient regions are sufficient to predict DTA with significant accuracy.

The left side of both tables, lists the performance drops achieved when only the salient regions in the input are used for prediction. In this setting, a threshold of $10\%ile$ means that $\frac{9}{10}^{th}$ of the input data is used for prediction since these are the salient features. On the right side of the tables, the performance drops achieved upon occlusion of the salient regions are listed. In this setting a threshold of 90% again means that $\frac{9}{10}^{th}$ of the input data is used for prediction.

Presenting the results thus allows us to easily make observations regarding trends and compare across the two settings. A further discussion of the results follows in Section 3.4.

3.4 Discussion

From the results in Tables 1 and 2, in line with our expectations, the performance drops are greater when the fraction of input used for prediction decreases ie. moving down the rows. Further, we can observe that occluding the same fraction of inputs results in much smaller performance drops when non-salient regions are occluded ie. in the left half of the tables. This indicates that the explanations we generate indeed identify the salient regions in the input successfully.

We note that the conformity measure ΔC defined in Section 2.3 shows consistently high values even for low quantile thresholds. Further, we can observe that ΔC increased as the salient region increases in fraction ie. moving down the rows on the right side of the tables. This is in line with the expectation that when the explanation sparisity decreases, all reactive regions of the drug and target would be identified by the explanations, resulting in a near perfect match.

3.5 Visualization of Saliency Maps

The generated saliency maps for the target protein sequences from Davis and KIBA datasets are depicted in Figures 1 and 2, while drug saliency maps for both datasets are shown in Figure 3.

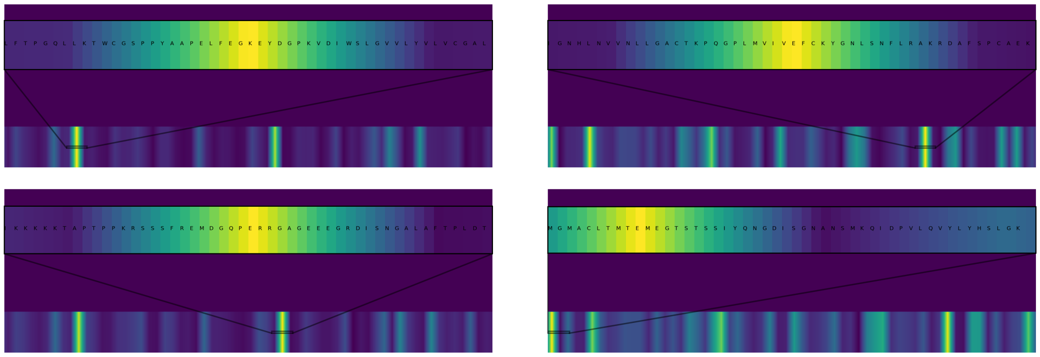

Figure 1: Saliency maps visualized for target protein sequences from the Davis dataset; Yellow regions are highly reactive; Most reactive regions are identified using a Gaussian-weighted Moving Average (GMA) and shown zoomed in with corresponding amino acid sequences

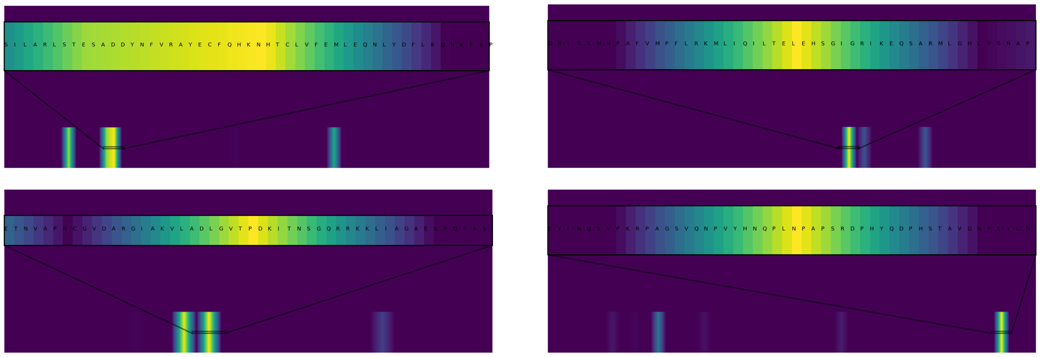

Figure 2: Saliency maps visualized for target protein sequences from the KIBA dataset; Yellow regions are highly reactive; Most reactive regions are identified using a Gaussian-weighted Moving Average (GMA) and shown zoomed in with corresponding amino acid sequences

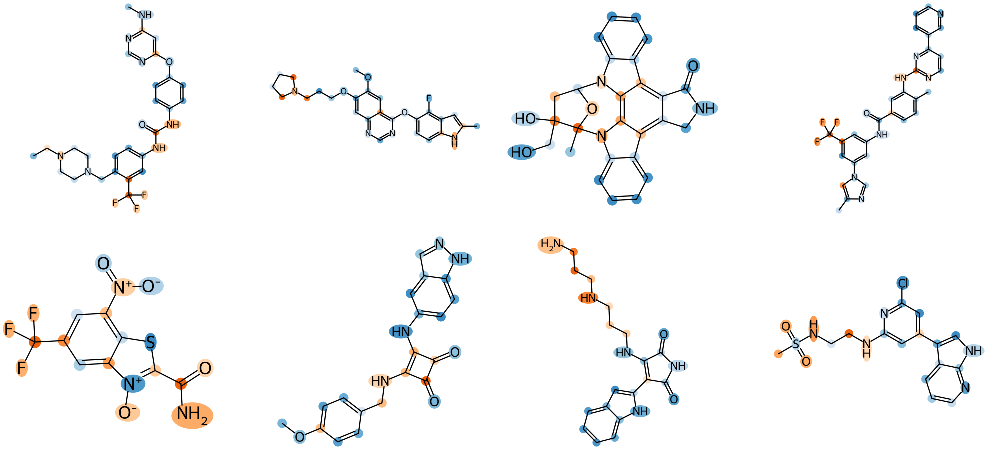

Figure 3: Saliency maps visualized for drug molecular graphs from the Davis (top) and KIBA (bottom) datasets; Red regions are highly reactive; Amines, amides, sulfonamides etc., are successfully identified and are structural alerts that correlate with specific toxicological endpoints

4 Conclusion

To the best of our knowledge this is the first work, which aims to study which parts of drugs and proteins interact with each other using the lens of XAI (eXplainable AI). We successfully generate drug and target saliency maps for GraphDTA type models trained for DTI prediction. We also conduct a quantitative study of the explanations generated thus, and propose a novel metric to quantify how well the drug and target saliency maps conform with each other.

4.1 Future Work

Considering the constraints of a course project, as well as significant computational constraints, we have identified a number of potential directions in which this work may be extended. The concept of paired explanations is relatively unexplored even in broader ML and XAI literature, with no existing metrics. The ΔC metric we propose is roughly designed and needs improvement. Further, experiments on MGraphDTA, as well as adapting explainability methods like GradCAM++, EigenCAM etc. were infeasible due to time and computational constraints.

Acknowledgements

We would like to thank Prof. Manikandan Narayanan for helpful discussions at crucial junctures during this course project. We also acknowledge and thank the authors of [21], the release of their code, the study of which was instrumental in the design of our algorithm in Section 2.2, as well as in preprocessing the raw SMILES strings and protein sequences to construct the drug molecular graph.

Work done after presentation

After the presentation, we redesigned the metric ΔC to more accurately reflect conformity, ie. made it a “higher is better" metric. To do so, we proved that the upper bound of the RMS of the differences between each pair of performance drops to be $\sqrt{\frac{2}{3}}$. We also carried out experiments to try adapting other explainability methods to our setting. These experiments were omitted from the report due to a lack of completeness.

Author Contributions

Both authors jointly contributed in equal parts to writing the report and preparing the presentation.

Vishnu formulated the key questions and hypotheses that guided the research. He implemented the code for training and testing the models and preprocessed the datasets. He also carried out hyperparameter tuning and performance evaluation and implemented visualization techniques for the drug and target saliency maps using `matplotlib`. He also helped develop the metric ΔC proposed in Section 2.3 and GradTAM proposed in Section 2.2.

Ankita conducted an in-depth review of existing research on GNNs and their applications for DTI predictions, gathering foundational papers and existing resources. She identified the key datasets (Davis, KIBA) and assisted in refining the scope, key questions and objectives of the project. She also contributed towards formulating the design and layout of visual explanations and helped identify and debug issues during development phases. She also assisted in selecting visualization tools and libraries for visualizing drug saliency maps.

Code Repository

The code used for this project is available open source on github at this repository. We also release pretrained models for the DTI prediction task, as well as results of the visualization of saliency maps for both drugs and targets.

References

[1] S. Bai, F. Zhang, and P. H. Torr. Hypergraph convolution and hypergraph attention. *Pattern Recognition*, 110: 107637, Feb. 2021. ISSN 0031-3203. doi: 10.1016/j.patcog.2020.107637. URL `http://dx.doi.org/10.1016/j.patcog.2020.107637`.

[2] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst. Geometric deep learning: going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017.

[3] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, Mar. 2018. doi: 10.1109/wacv.2018.00097. URL `http://dx.doi.org/10.1109/WACV.2018.00097`.

[4] M. Fey and J. E. Lenssen. Fast graph representation learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019.

[5] K. Y. Gao, A. Fokoue, H. Luo, A. Iyengar, S. Dey, P. Zhang, et al. Interpretable drug target prediction using deep neural representation. In *IJCAI*, volume 2018, pages 3371–3377, 2018.

[6] K. Huang, T. Fu, W. Gao, Y. Zhao, Y. H. Roohani, J. Leskovec, C. W. Coley, C. Xiao, J. Sun, and M. Zitnik. Therapeutics data commons: Machine learning datasets and tasks for drug discovery and development. In *NeurIPS Datasets and Benchmarks*, 2021. URL `https://api.semanticscholar.org/CorpusID:237264044`.

[7] 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*, 10(35):20701–20712, 2020.

[8] A. Khan and E. B. Mobaraki. Interpretability methods for graph neural networks. In *2023 IEEE 10th International Conference on Data Science and Advanced Analytics (DSAA)*, pages 1–4. IEEE, 2023.

[9] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL `https://openreview.net/forum?id=SJU4ayYgl`.

[10] M. B. Muhammad and M. Yeasin. Eigen-cam: Class activation map using principal components. In *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE, July 2020. doi: 10.1109/ijcnn48605.2020.9206626. URL `http://dx.doi.org/10.1109/IJCNN48605.2020.9206626`.

[11] T. Nguyen, H. Le, T. P. Quinn, T. Nguyen, T. D. Le, and S. Venkatesh. GraphDTA: predicting drug–target binding affinity with graph neural networks. *Bioinformatics*, 37(8):1140–1147, 10 2020. ISSN 1367-4803. doi: 10.1093/bioinformatics/btaa921. URL `https://doi.org/10.1093/bioinformatics/btaa921`.

[12] D. Omeiza, S. Speakman, C. Cintas, and K. Weldermariam. Smooth grad-cam++: An enhanced inference level visualization technique for deep convolutional neural network models, 2019.

[13] H. Öztürk, A. Özgür, and E. Ozkirimli. Deepdta: deep drug–target binding affinity prediction. *Bioinformatics*, 34 (17):i821–i829, 2018.

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

[15] B. A. Plummer, M. I. Vasileva, V. Petsiuk, K. Saenko, and D. Forsyth. Why do these match? explaining the behavior of image similarity models. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, pages 652–669. Springer, 2020.

[16] P. E. Pope, S. Kolouri, M. Rostami, C. E. Martin, and H. Hoffmann. Explainability methods for graph convolutional neural networks. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10764–10773, 2019. URL `https://api.semanticscholar.org/CorpusID:198904065`.

[17] S. Poppi, M. Cornia, L. Baraldi, and R. Cucchiara. Revisiting the evaluation of class activation mapping for explainability: A novel metric and experimental analysis. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2299–2304, 2021. URL `https://api.semanticscholar.org/CorpusID:233324270`.

[18] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[19] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[20] H. N. T. Tran, J. J. Thomas, and N. H. A. H. Malim. Deepnc: a framework for drug-target interaction prediction with graph neural networks. *PeerJ*, 10:e13163, 2022.

[21] Z. Yang, W. Zhong, L. Zhao, and C. Yu-Chian Chen. Mgraphdta: deep multiscale graph neural network for explainable drug–target binding affinity prediction. *Chem. Sci.*, 13:816–833, 2022. doi: 10.1039/D1SC05180F. URL `http://dx.doi.org/10.1039/D1SC05180F`.

[22] J. Zhang, S. A. Bargal, Z. Lin, J. Brandt, X. Shen, and S. Sclaroff. Top-down neural attention by excitation backprop. *International Journal of Computer Vision*, 126(10):1084–1102, Dec. 2017. ISSN 1573-1405. doi: 10.1007/s11263-017-1059-x. URL `http://dx.doi.org/10.1007/s11263-017-1059-x`.

[23] Y. Zhang, C. Liu, M. Liu, T. Liu, H. Lin, C.-B. Huang, and L. Ning. Attention is all you need: utilizing attention in AI-enabled drug discovery. *Briefings in Bioinformatics*, 25(1):bbad467, 01 2024. ISSN 1477-4054. doi: 10.1093/bib/bbad467. URL `https://doi.org/10.1093/bib/bbad467`.

[24] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.