Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Manuscript Category

Counterfactual Explanation with Multi-Agent Reinforcement Learning for Drug Target Prediction

Tri Minh Nguyen [1,*], Thomas P Quinn [1], Thin Nguyen [1] and Truyen Tran [1,]

[1] Applied Artificial Intelligence Institute, Deakin University, Victoria, Australia

*To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Several accurate deep learning models have been proposed to predict drug-target affinity (DTA). However, all of these models are black box hence are difficult to interpret and verify its result, and thus risking acceptance. Explanation is necessary to allow the DTA model more trustworthy. Explanation with counterfactual provides human-understandable examples. Most counterfactual explanation methods only operate on single input data, which are in tabular or continuous forms. In contrast, the DTA model has two discrete inputs. It is challenging for the counterfactual generation framework to optimize both discrete inputs at the same time.
Results: We propose a multi-agent reinforcement learning framework, Multi-Agent Counterfactual Drug-target binding Affinity (MACDA), to generate counterfactual explanations for the drug-protein complex. Our proposed framework provides human-interpretable counterfactual instances while optimizing both the input drug and target for counterfactual generation at the same time. The result on the Davis dataset shows the advantages of the proposed MACDA framework compared with previous works.
Availability: The Python implementation is available at https://github.com/ngminhtri0394/MACDA
Contact: minhtri@deakin.edu.au

1 Introduction

Drug-target binding affinity (DTA) prediction is an important step in drug discovery and drug repurposing (Thafar *et al.*, 2019). Many high-performance DTA models have been proposed, but they are mostly black-box and thus lack human interpretability (Öztürk *et al.*, 2018, 2019; Torng and Altman, 2019; Zheng *et al.*, 2020; Jiang *et al.*, 2020; Nguyen *et al.*, 2020; Tri *et al.*, 2020). This lack of interpretability makes it difficult to use deep learning models to *distill knowledge* about how drugs bind to their targets. Explainable AI methods explain how a model works, and therefore facilitate knowledge distillation. When applied to DTA prediction, model explanations could produce insights that feedback into the research pipeline and inform chemical drug synthesis.

Counterfactual explanation is one popular approach to explaining the behaviour of a deep neural network, which works by systematically answering the question "How would the model output change if the inputs were changed in this way?". Research into this approach to explainable AI has mostly focused on image data and tabular data (Vermeire and Martens, 2020; Mothilal *et al.*, 2020; Dhurandhar *et al.*, 2018; Cheng *et al.*, 2020). In the context of the drug-target affinity, counterfactual explanations are not widely used. Instead, explanation methods rely on feature attribution scores and gradients (Preuer *et al.*, 2019; Pope *et al.*, 2019; McCloskey *et al.*, 2019), which may fail to capture high-order interactions between features (Tsang *et al.*, 2018). It remains an open problem of how to produce counter-factual explanations into drug-target affinity models.

There are two key challenges in extending counterfactual explanations to drug-target affinity models. First, the inputs to a DTA model, represented most often as sequences or graphs, are discrete not continuous. Therefore, gradient-based counterfactual generation methods, which operate on the continuous data, cannot be applied. Meanwhile, gradient-free combinatorial methods like *in silico mutagenesis* (Zhou and Troyanskaya, 2015) are computationally expensive to run. Second, DTA models have two distinct inputs, the drug and the target. Changes to both the drug molecule and protein can influence the binding affinity, either separately or jointly. Generating counterfactuals for drug and target separately may not lead to an optimal solution (see Fig. 1).

We propose **M**ulti-**A**gent **C**ounterfactual **D**rug-target binding **A**ffinity (**MACDA**) framework, which uses multi-agent reinforcement learning (MARL) to solve both challenges. Firstly, MARL solves the challenge of discrete inputs as they are naturally designed for discrete action spaces. In the DTA problem, adding or removing bonds and atoms on drug can be thought of as a discrete action space. Likewise, adding or removing motifs or residues on a protein can also be thought of as a discrete action space.

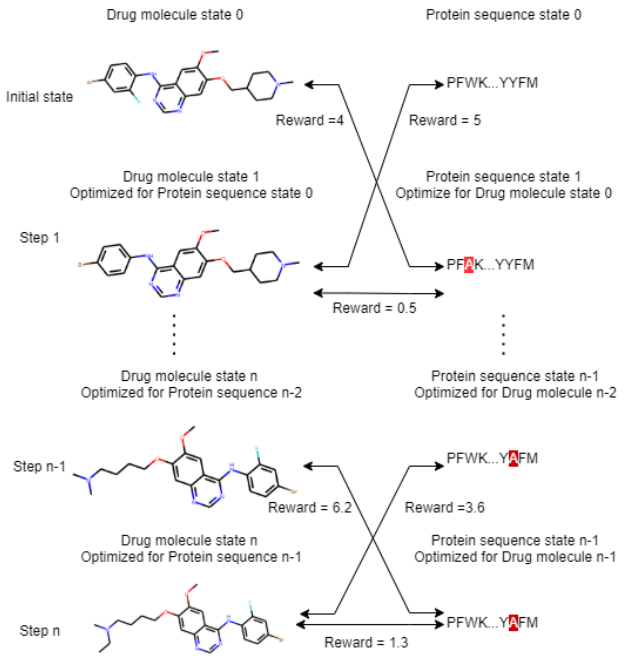

Fig. 1. The unstable environment problem in the drug-target binding affinity counterfactual generation. The reward indicates how good the drug-target counterfactual. After step 1, as the drug molecule state 1 is optimized for protein sequence state 0, drug state 1 - protein state 0 pair has high reward. However, after step 1, the protein sequence is changed to state 1 which is the optimal sequence for drug molecule state 0. As a result, the drug state 1 is not the optimal molecule for the current protein sequence. The cycle continues and both drug and protein sequence never reach their optimal state with respect to each other. Best viewed in color.

Secondly, MARL solves the challenge of multiple inputs because the drug and the protein can be represented as different action space inputs within a single model.

In summary, we propose the multi-agent reinforcement learning framework, MACDA, for the drug-target binding affinity counterfactual generation problem. We evaluate the proposed framework on the publicly available Davis dataset, where we observe a stable learning process and counterfactual explanations that agree with the state-of-the-art method in molecule counterfactual generation.

2 Related works

2.1 Drug-target affinity models

Drug-target binding affinity measured by a disassociation constant K_d indicates the strength of the binding force between the target protein and its ligand (drug or inhibitor). Drug-target binding affinity prediction methods can be categorized into two main approaches: structural approach and non-structural approach (Thafar *et al.*, 2019). The structural approach (Meng *et al.*, 1992; Jorgensen *et al.*, 1983; Pullman, 2013; Raha *et al.*, 2007) uses the 3D information of the protein structure and ligand to run a drug-target interaction simulation. On the other hand, the non-structural approach (Nguyen *et al.*, 2020; Öztürk *et al.*, 2018, 2019; Tri *et al.*, 2020) uses other information such as protein sequence, atom valence, hydrophobic, and others to apply the machine learning models that are trained from existing databases to predict the binding affinity. The former approach makes extensive use of explicit domain knowledge to build an accurate simulator. The latter approach, on the other hand, implicitly extracts the knowledge hidden in the data itself and instills the knowledge in the prediction model. In recently, highly accurate models are typically based on deep neural networks, which are mostly black-box. This poses a great question of how to explain the behaviours of the deep models, and distill the explicit knowledge from them. In MACDA, we take the non-structural approach.

2.2 Explaining deep neural networks for DTA

Explaining the deep learning model prediction on the drug molecule properties has been studied recently. Within the scope of this paper, we briefly review two major approaches: feature attribution and graph-based methods.

Feature attribution measures the relevance score of the input feature with respect to the predicted affinity score y, either using gradient (Preuer *et al.*, 2019; Pope *et al.*, 2019) or surrogate model (Rodríguez-Pérez and Bajorath, 2019). Gradient-based methods take advantage of the derivative of the output with respect to the input. (McCloskey *et al.*, 2019) use integrated gradients on graph convolution model trained on the molecular binding synthesis dataset to analyze the binding mechanism. However, gradient-based methods could be misleading or prone to gradient saturation (Ying *et al.*, 2019). Surrogate-based methods generate a surrogate explanatory model g which is interpretable (linear or decision tree) and can approximate the original function f.

Graph-based methods are suitable for DTA because the structure of the drug molecule can be represented naturally with the graph structure. The graph can be explained by subsets of edges and node features which are important for model f prediction of class c. For example, GNNExplainer (Ying *et al.*, 2019) finds a subgraph G' of input graph G, and subfeature X' of input feature X which maximizes the mutual information between $f(G, X)$ and $f(G', X')$, but is argued to not generalize well (Numeroso and Bacciu, 2020). Attention-based graph neural network (Veličković *et al.*, 2018; Shang *et al.*, 2018; Ryu *et al.*, 2018) is a mechanism that can facilitate explanation such as the influence of substructure to the solubility property (Shang *et al.*, 2018), visualizing the importance of neighbor nodes via an attention score.

2.3 Counterfactual explanations

Instead of assigning the relevance score for the input features to the model prediction, counterfactual explanation finds the perturbed instance with minimal change while maximizing the difference in the model prediction outcome. In the early work, (Wachter *et al.*, 2017) generate the counterfactual instance by optimizing the following formula:

$$L(x, x', y', \lambda) = \lambda \cdot (f(x) - y')^2 + d(x, x') \tag{1}$$

where y' is the user's desired model prediction, $d(x, x')$ is the Manhattan distance between instance x and counterfactual instance x'. The drawback of optimizing Eq. (1) is that it does not ensure the number of changed features is the smallest and the new feature value is in the data distribution. To ensure that the counterfactual instance follows the data distribution, (Dhurandhar *et al.*, 2018) utilize the autoencoder trained on training data as the prototype. (Mothilal *et al.*, 2020) propose counterfactual framework DiCE which generates diverse counterfactual samples while maintaining sparsity and proximity.

2.4 Reinforcement learning: Single and Multi-agent

Reinforcement learning (RL) is the process of agent learning to find the optimal action for situations that maximizes the long-term rewards (Sutton and Barto, 2018). For the single agent case, an agent interacts with the environment. At each time step t, the agent observes environment state $s_t \in S$ and chooses an action $a_t \in A$ using its policy $\pi(a_t|s_t)$. By completing the action, the agent receives a reward r_t and changes the environment to the next state s_{t+1}.

There are two main approaches to RL: value-based and policy-based. The value-based methods estimate the value function for each state:

$$v_\pi(s) = \mathbb{E}[R_t|s_t = s]; \quad R_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k} \tag{2}$$

where γ is the discount factor. The action is chosen based on the action value:

$$q_\pi(s, a) = \mathbb{E}[R_t|s_t = s, a_t = a] \tag{3}$$

The policy-based methods, on the other hand, optimize the agent's policy $\pi(a|s, \theta)$ where θ is the parameter. The two methods can be combined, e.g., both value function and the policy are estimated, as in the celebrated actor-critic methods (Konda and Borkar, 1999; Morimura *et al.*, 2009).

Multi-agent reinforcement learning (MARL) is the generalization from a single agent to multiple agents that share the same environment. Each agent interacts with the environment and with other agents. The challenge of multi-agent RL is finding the optimal policy for each agent with respect not only to the environment but also to other agent's policies. Many approaches solving the multi-agent setting have been proposed, ranging from cooperative communication (Tan, 1993; Fischer *et al.*, 2004) to competitive environment (Littman, 1994; Perez-Liebana *et al.*, 2019).

2.5 Using reinforcement learning for explanations

Reinforcement learning has been applied to generate explanations. (Hendricks *et al.*, 2016) use RL-based loss and image captioning to explain image classification model. (Li *et al.*, 2016) learn erasing a minimum set of words in a sentence that changes the model output. (Numeroso and Bacciu, 2020) generate a counterfactual explanation for a molecule using MEG framework, a multi-objective reinforcement learning, to maximize the prediction model output change and the similarity between original and counterfactual molecule instance. We use the multi-agent version of the MEG framework as the baseline for our experiments. Compared to multi-agent MEG framework, our proposed MACDA framework uses multi-agent actor-critic approach in which the value function takes account of both agents observations.

3 Methods

3.1 Preliminaries

To provide the necessary technical background we briefly introduce Actor-Critic Reinforcement learning (Sec. 3.1.1), and the general multi-agent reinforcement learning (MARL) (Sec. 3.1.2).

3.1.1 Actor-Critic RL

Actor-critic In this learning framework, the agent learns to maximize the expected discount returns over the future T steps: $J = \mathbb{E}\left[\sum_{t=1}^{T} \gamma^t r^t\right]$ for discount factor $\gamma \in [0, 1]$. To optimize the agent policy with respects to the expected discount returns, the gradient is estimated as:

$$\nabla_\theta J(\pi_\theta) = \nabla_\theta \log \pi_\theta(a_t|s_t) R(t) \tag{4}$$

where $R(t) = \sum_{t'=T}^{\infty} \gamma^{t'-t} r^{t'}(s^{t'}, a^{t'})$. However, due to the high variance of $R(t)$, the critic function $Q_\psi(s_t, a_t)$ is used instead.

$$Q_\psi(s_t, a_t) = \mathbb{E}_{a_1 \sim \pi_1, \ldots, a_N \sim \pi_N, s \sim T}[R(t)] \tag{5}$$

The gradient $\nabla_\theta J(\pi_\theta)$ is replaced as:

$$\nabla_\theta J(\pi_\theta) = \nabla_\theta \log(\pi_\theta(a_t|s_t)) Q_\psi(s_t, a_t) \tag{6}$$

The $Q_\psi(s_t, a_t)$ function is learned by minimizing the TD loss:

$$\mathcal{L}_Q(\psi) = \mathbb{E}_{(s,a,r,s') \sim D}\left[(Q_\psi(s, a) - y)^2\right] \tag{7}$$

$$y = r(s, a) + \psi \mathbb{E}_{a' \sim \pi(s')}[Q_{\tilde{\psi}}(s', a')] \tag{8}$$

where y is the target, $Q_{\tilde{\psi}}$ is the target Q-value function which is the value of previous Q-functions stored in the replay buffer.

Soft Actor-Critic To avoid converging to the local minimum, the entropy term $-\alpha \log \pi_\theta(a|s)$ is incorporated into the policy gradient:

$$\nabla_\theta J(\pi_\theta) = \mathbb{E}_{s \sim D, a \sim \pi}\left[\nabla_\theta \log \pi_\theta(a|s) Q^*(a, s)\right]$$

$$Q^*(a, s) = Q_\psi(s, a) - \alpha \log \pi_\theta(a|s) - b(s)$$

where $b(s)$ is a state-dependent baseline, $\alpha > 0$ is trade-off coefficient. Then the target y in Eq. (7) is updated as:

$$y = r(s, a) + \psi \mathbb{E}_{a' \sim \pi(s')}\left[Q_{\tilde{\psi}}(s', a') - \alpha \log \pi_{\tilde{\theta}}(a'|s')\right] \tag{9}$$

where $\tilde{\psi}$ and $\tilde{\theta}$ are the network parameters of the target critics and policies.

3.1.2 Multi-agent reinforcement learning (MARL)

The multi-agent reinforcement learning framework is described as Markov Decision game with a tuple $< N, S, \{O_i\}_{i \in N}, \{A_i\}_{i \in N} >, P, \{R_i\}_{i \in N} >$, where N is the number of agent, S is the set of state, $\{O_i\}_{i \in N}$ is the observation space of N agents, $\{A_i\}_{i \in N}$ is joint action of N agents, and $\{R_i\}_{i \in N}$ is reward for each agents. The agent i^{th} chooses an action based on the policy function $\pi_{\theta_i} : O_i \to P(A_i)$ where $P(A_i)$ is the distribution over action set A_i given observation O_i. Then the state s and joint action A lead to the next state s' with the probability function P. The agent i^{th} receives a reward based on the state and action of other agents $R_i : S \times \{A_i\}_{i \in N}$.

3.2 Drug-protein counterfactual generation with MARL

We now describe our MARL framework for generating drug-target counterfactuals. MARL is particularly suitable because it works naturally on multiple discrete action spaces. In our setting, the two action spaces correspond to the discrete modifications in the molecule space and protein space, respectively. In particular, we will employ a MARL framework known as Multi-agent Actor-Attention-Critic (MAAC) ((Iqbal and Sha, 2019)). This framework is flexible, easy to train, and is natural for exploring the joint space of protein-drug complex. MAAC allows separated policies for drug and protein, but with the common critics and rewards. It uses the attention mechanism to dynamically select relevant information shared by the other agent, and this agrees well with the selective binding mechanism often found in the protein-drug complex (Tri *et al.*, 2020).

The framework is illustrated in Fig. 2. There are two agents, one responsible to generate counterfactuals for the protein, and the other for the drug. The two agents work in tandem to produce the joint counterfactuals for the protein-drug complex. Each agent has its own Q-value function. Two agents communicate through the reward function described in Eq. 12 and calculating Q-value function (Eq. 10).

In what follows, we describe the framework components. In particular, the action space for drug and protein are provided in Sec. 3.2.1 and Sec. 3.2.2, respectively. The overall reward is presented in Sec. 3.2.3.

3.2.1 Drug counterfactual generation

We adopt the drug molecule generation strategy in Mol-DQN ((Zhou *et al.*, 2019)). There are three action categories: (a) **Add atom**: Given an admissible set of atoms $\mathcal{E} = \{Atom_1, .., Atom_N\}$, one atom $Atom_i$ is inserted into the drug molecule at a time. Then a bond is formed between the newly added atom and a position satisfying the valence constraint. Therefore,

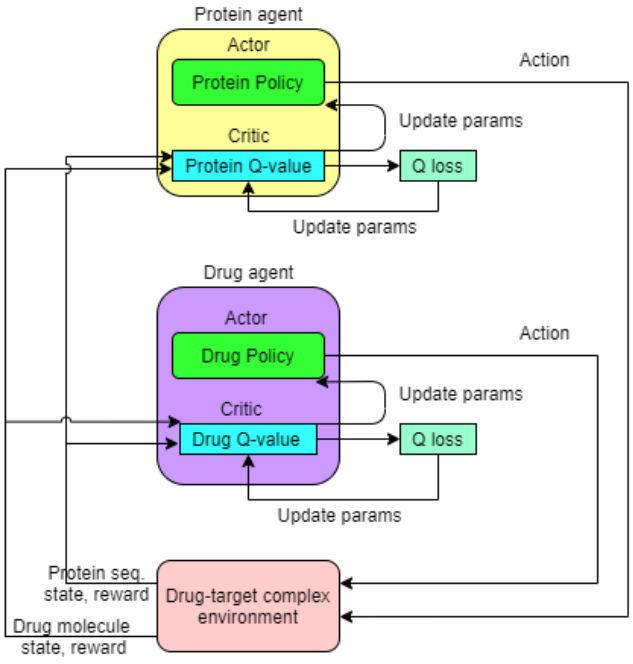

Fig. 2. The overview of MARL framework for drug-target counterfactual generation. Two agents, protein and drug, have their own actor and critic. The actor chooses the action using its policy. Two agents take actions and the drug-target environment returns the reward, which is a sum of prediction change and protein/drug similarity, and the new state of drug and protein. Then the critic calculates its Q value using the reward value. The Q value is used to update the critic network.

given n_p positions satisfying the constraint, there are n_p instances generated. (b) **Add bond**: One or more bond is added up to triple bond between two atoms with free valence. (c) **Remove bond**: One or more bond is removed from the existing bond. If there is no bond between two atoms after removal, then the disconnected atom is removed.

All possible actions are generated and used as action space in the reinforcement learning framework described in Sec. 3.2.3.

3.2.2 Protein counterfactual generation

We apply the alanine scanning (Gray *et al.*, 2017) for the protein counterfactual generation. Alanine is widely used to determine the contribution of protein residues in the protein function or drug-protein binding (Gray *et al.*, 2017). Alanine is chosen as its size is not too large which avoids steric hindrance. In addition, the methyl function group allows it to mimic the secondary structure of the residues it replaces ((Gray *et al.*, 2017)). For the protein sequence $P = r_i, i \leq l$ where l is protein sequence length, a single residue r_i is replaced with alanine to create a single point alanine mutation. All possible single point mutations are generated and used as action space.

3.2.3 Multi-agent actor-attention-critic for counterfactual generation

The idea of multi-agent actor-critic is that the Q-value function of agent i is calculated based on the observation of other agents $s = (s_1, ..., s_N)$:

$$Q_i^\theta = f_i\left(g_i(s_i, a_i), x_i\right) \tag{10}$$

where f_i and g_i are multi-layer perceptron, x_i is the weighted sum of other agents value:

$$x_i = \sum_{j \neq i} \alpha_{ij} \sigma\left(V g_j(s_j, a_j)\right) \tag{11}$$

where V is the linear transformation, σ is Leaky ReLU, and α_{ij} is the attention score computed as in (Vaswani *et al.*, 2017) taking g_i, g_j as the inputs. Learning in this actor-critic framework then proceeds for each agent using the framework described in Sec. 3.1.1.

In our context of protein-drug counterfactual generations, this boils down to using state-action function of drug agent to influence the Q-function of the protein agent and vice versa.

Multi-objective reward function To find the counterfactual satisfying two constraints: maximizing the change in the predicted binding affinity and maximizing the similarity between original instance and counterfactual instance, the reward function is defined as:

$$\begin{aligned} R(s) = \alpha_r \text{sgn}\left(\|\mathcal{F}(P', D') - \text{GT}_{F,D}\|_1 - \|\mathcal{F}(P, D) - \text{GT}_{F,D}\|_1\right) \times \\ \times \|\mathcal{F}(P', D') - \mathcal{F}(P, D)\|_1 \\ + \alpha_p \text{SIM}(\mathcal{F}_e(P), \mathcal{F}_e(P')) + \alpha_d \text{SIM}(\mathcal{F}_e(D), \mathcal{F}_e(D')) \end{aligned} \tag{12}$$

where P' and D' are the counterfactual instance of drug D and protein P, $\text{GT}_{F,D}$ is the ground truth binding affinity value, $\mathcal{F}_e$ is the encoded representation of drug or protein in the DTA model, α_r, α_d. and α_p are coefficient to balance between the predicted affinity change and the similarity. The similarity is the cosine similarity:

$$\text{SIM}(\mathcal{F}_e(x), \mathcal{F}_e(y)) = \frac{\mathcal{F}_e(x) \cdot \mathcal{F}_e(y)}{\|\mathcal{F}_e(x)\| \|\mathcal{F}_e(y)\|}. \tag{13}$$

For the sign function term, as we include the similarity term in the reward, the model is likely to generate toward the original instance. Therefore, we add the sign function to give negative reward when the generated molecules move toward the original instance. We add both protein and drug molecule similarity terms to the reward. First, both similarity terms help the model to generate molecules and sequences with minimal change, satisfying one of counterfactual constraints. Second, it works as a communication between two agents where the drug agent searches for a molecule that does not require significant change in protein and vice versa. The second term, $\|\mathcal{F}(P', D') - \mathcal{F}(P, D)\|_1$, is to meet the model output change constraint. The two similarity terms encourage the similarity between original instance and counterfactual instance.

3.3 Experiments

3.3.1 Dataset

We evaluate our method MACDA on the Davis dataset (Davis *et al.*, 2011). Davis dataset is the drug-target binding affinity of 442 target proteins and 72 drugs. The K_D (kinase dissociation constant) metric is used to measure the binding affinity between the target protein and the drug molecule. The drug-target pairs between Tyrosine-protein kinase ABL1 (Human) and 50 drugs in the training set are chosen to generate counterfactual instances.

3.3.2 Baseline

We extend the molecule counterfactual generation MEG framework (Numeroso and Bacciu, 2020) to the drug-target counterfactual generation task. As the MEG framework only has a single agent handling the optimization for drug molecule, we add another agent handling the protein sequence optimization. The protein agent has action space described in Sec. 3.2.2. The protein agent calculates and updates its Q-function in the same manner as the drug agent. Two agents work independently to optimize the common reward function (see Eq. (12)).

3.3.3 Implementation detail

The MACDA framework is implemented in Python using Pytorch. GraphDTA-GCNNet (Nguyen *et al.*, 2020) is used as a drug-target binding affinity prediction model because of its simplicity and high performance. GraphDTA-GCNNet receives the drug molecule graph and protein sequence

as the inputs. The drug molecule observation in MACDA framework is the drug fingerprint. The protein observation in MACDA framework is the alphabet sequence encoded to integer sequence. The protein sequence length is fixed at 1000 residues.

For each drug-target instance, the top ten counterfactual instances with the highest reward are chosen. The hyperparameters in the experiment are shown in Table 1.

Table 1. The hyper-parameters used in the experiments

Hyper parameters	Value
γ	0.99
Batch size	1024
Policy learning rate	0.001
Critic learning rate	0.001
Number of episode	10000

3.3.4 Evaluation metrics

Two methods are evaluated using five metrics: average reward, average drug encoding similarity, average protein encoding similarity, average mean absolute error (MAE) between the predicted binding affinity of original and counterfactual instance, and drug-likeness (QED). The reward score is defined in Eq. (12). The similarity score is defined in Eq. (13).

The first four metrics are the objectives optimized by the proposed framework while the QED assesses the validity of the drug counterfactual instance.

3.3.5 Quantitative results

Table 2 shows the quantitative result. Both methods have the same average protein sequence encoding. As the alanine scanning only changes a single residue to alanine, the encoded representation change is small. The proposed method shows advantages in reward, MAE, drug similarity, and QED. This is further elaborated in the Figs. 3 - 6. Due to the constant change in both drug molecule (see Fig. 5) and protein sequence (see Fig. 6), it is difficult for multi-agent MEG to optimize the common objective. There is no communication or common strategy for two MEG agents. On the other hand, there is communication between two agents of MACDA framework in calculating the Q-value (Eqs. (10-11)) so the learning process is more stable (Fig. 3).

Table 2. The average reward, average MAE, drug encoding similarity, target encoding similarity

Method	Avg. Reward↑	Avg. MAE↑	Avg. Drug Sim.↑	Avg. Protein Sim. ↑	QED ↑
MACDA	0.5911	0.3582	0.8813	0.9995	0.4048
MA-MEG	0.5575	0.3065	0.8686	0.9995	0.3854

3.3.6 Analyze and visualize results

In this section, we analyze and visualize the result of the counterfactual instance generation to gain insights of the proposed framework and the explained DTA model.

Table 3 and Fig. 7 show the top three drug molecule counterfactuals of the ABL1-Imatinib pair ranked by the reward. All three counterfactual instances are created by adding/substituting the hydroxyl group. Adding hydroxyl group can increase the binding affinity by forming hydrogen-bond networks or decrease the affinity because of the high desolvation penalty (Cramer *et al.*, 2019). In this case, the DTA model decides that the appearance of the hydroxyl group decreases the affinity.

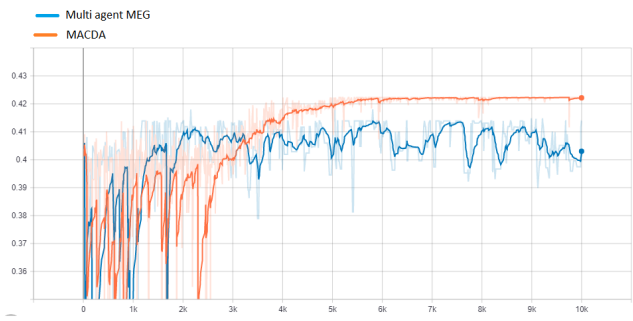

Fig. 3. The average reward of MA-MEG and MACDA over 10000 epochs. Figure best viewed in color.

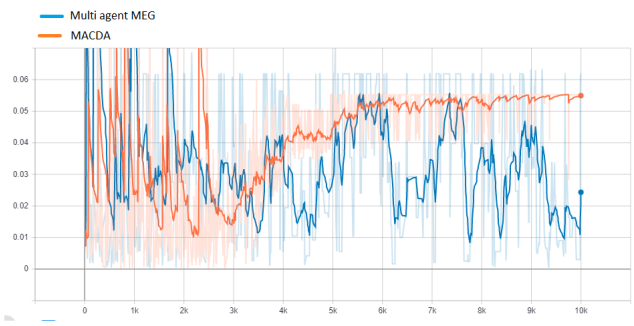

Fig. 4. The average prediction distance of MA-MEG and MACDA over 10000 epochs. Figure best viewed in color.

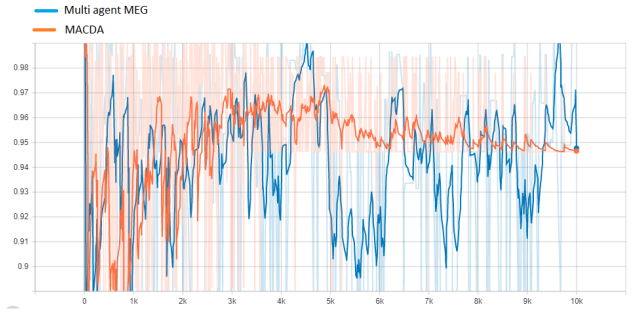

Fig. 5. The average drug molecule encoding similarity of MA-MEG and MACDA over 10000 epochs. Figure best viewed in color.

Table 3. The top three drug molecule counterfactual instance of Imatinib-ABL1 pair, the prediction value change and the similarity between counterfactual instance and original instance.

Molecule	Prediction value change	Similarity
M0 (Fig. 7b)	-0.2258	0.9604
M1 (Fig. 7c)	-0.2223	0.9637
M2 (Fig. 7d)	-0.2195	0.9422

We measure the frequency of each mutation point in the protein counterfactual instance over 500 counterfactual instances (see Fig. 8). There are four common mutation points: THR.136, TYR.147, LYS.236, and VAL.256. All four residues have exposed solvent accessibility (see Figs. 9 and 10) which is one of the conditions allowing drug-target interaction. VAL.256 is the active site of the ABL1-Imatinib (see Fig. 11).

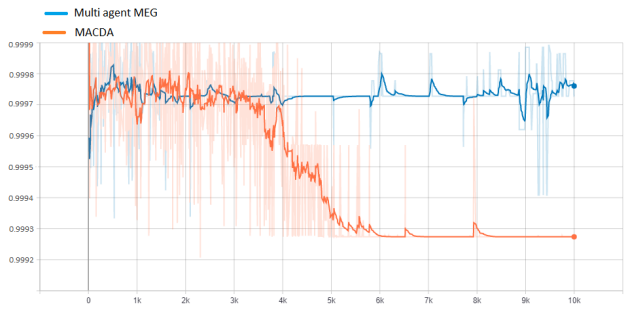

Fig. 6. The average protein sequence encoding similarity of MA-MEG and MACDA over 10000 epochs. Figure best viewed in color.

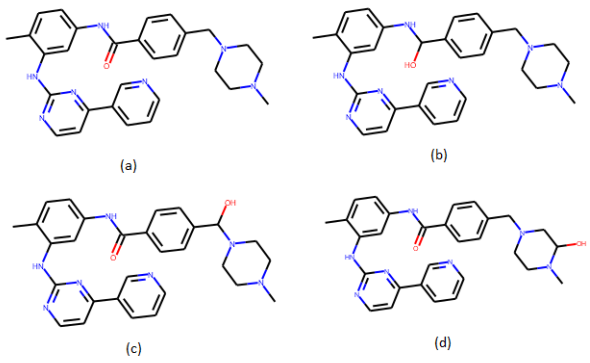

Fig. 7. (a) The original Imatinib molecule, (b)-(d) the top three counterfactual instances of ABL1-Imatinib complex.

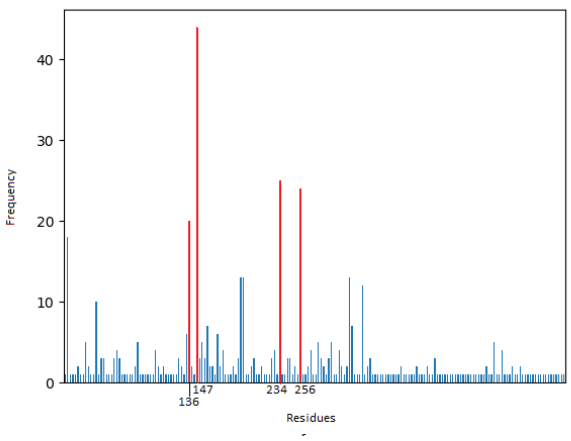

Fig. 8. The mutation point distribution over 500 counterfactual instances of 50 ABL1-drugs pairs.

In 500 counterfactual instances of ABL1-drug pairs, only the pairs with the binding affinity equal 5.0 (no interaction) ground-truth value have counterfactual instances that has positive change to the binding affinity. On the other hand, the pairs with binding affinity greater than 5.0 (having drug-target interaction) have no counterfactual instance having positive change. It may be more difficult to suggest changes that increase affinity in the case of bound drug-target pairs.

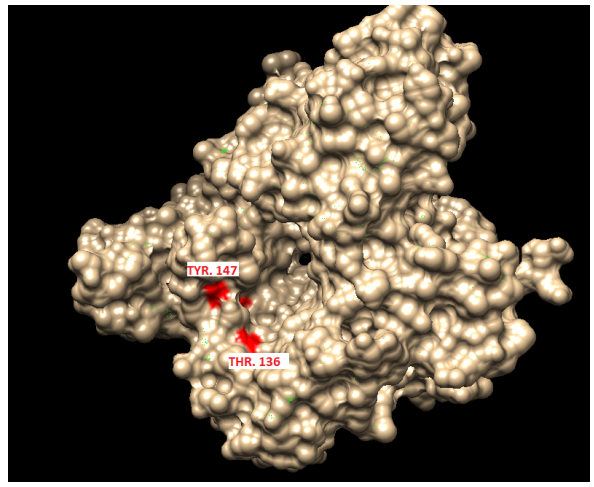

Fig. 9. Visualization of residue THR.136 and TYR.147 of protein ABL1 (PDB 5MO4). Figure best viewed in color.

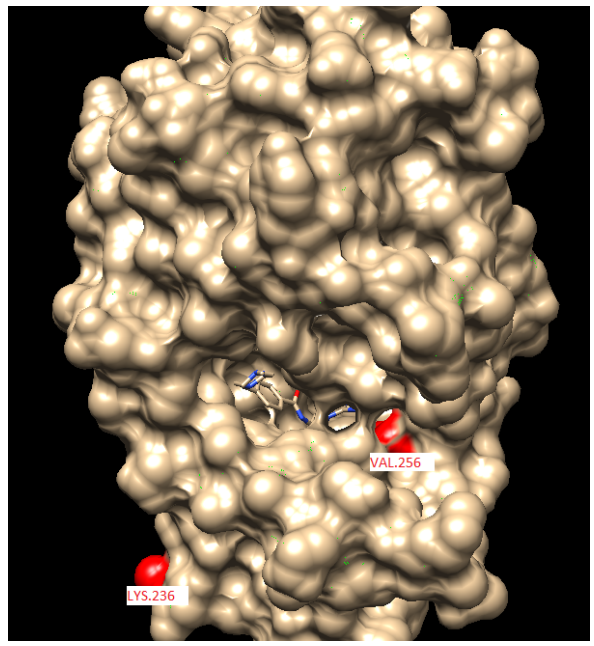

Fig. 10. Visualization of residue LYS.236 and VAL.256 of protein ABL1 (PDB 2HYY). Figure best viewed in color.

4 Conclusion

We have proposed a multi-agent reinforcement learning framework named MACDA (Multi-Agent Counterfactual Drug-target binding Affinity) to generate counterfactual explanations for the drug-target binding affinity model. To address the discrete molecule graphs and protein sequences, we use reinforcement learning to generate the counterfactual instances which maximize the change in the binding affinity and the similarity between counterfactual instances and original instances. To address the two-input problem of drug-target binding affinity prediction model, we use multi-agent reinforcement learning. Our multi-agent RL framework consists of a protein agent and a drug molecule agent. Both agents cooperate with each other to optimize the common multi-objective reward. The experiments show that the proposed framework provides a stable learning process and

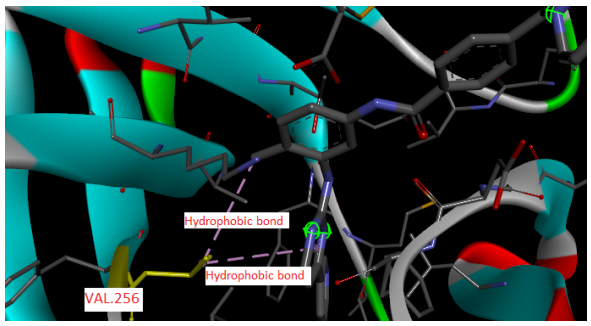

Fig. 11. The hydrophobic bond between VAL.256 of ABL1 and the flag-methyl group of Imatinib. Figure best viewed in color.

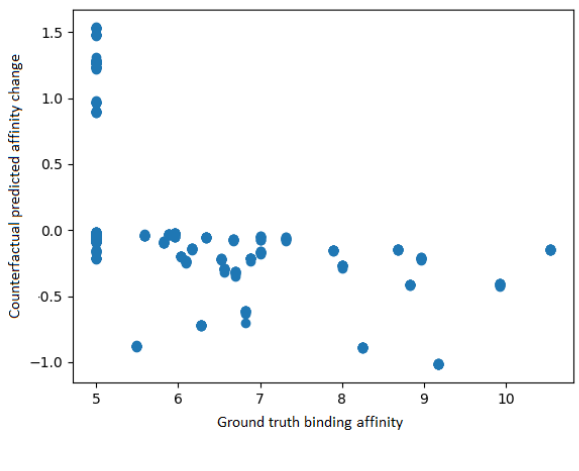

Fig. 12. The binding affinity differences between the original and counterfactual instances are plotted against the ground truth of the binding affinity of the original instance.

generates the counterfactual instances which maximize the binding affinity change and minimize the change in the input.

This work opens opportunities for future works. The counterfactual instances for protein sequences are limited to the alanine scanning process which is single point alanine mutation. Generating all possible multiple points mutation for protein sequences is challenging due to the large action space. Furthermore, the binding between drug and target protein does not simply rely on a single residue but a subset of residues around the binding pocket. Instead of scanning using a single residue, the protein sequence motif is also an interesting direction. For the drug molecules, the counterfactual instances can be generated at the fragment level instead of the atom level to maintain the drug-likeness of the molecules.

References

Bickerton, G. R. *et al.* (2012). Quantifying the chemical beauty of drugs. *Nature Chemistry*, **4**(2), 90–98.

Cheng, F. *et al.* (2020). DECE: Decision Explorer with Counterfactual Explanations for Machine Learning Models. *IEEE Transactions on Visualization and Computer Graphics*.

Cramer, J. *et al.* (2019). Hydroxyl Groups in Synthetic and Natural-Product-Derived Therapeutics: A Perspective on a Common Functional Group. *Journal of Medicinal Chemistry*, **62**(20), 8915–8930.

Davis, M. I. *et al.* (2011). Comprehensive analysis of kinase inhibitor selectivity. *Nature Biotechnology*, **29**(11), 1046–1051.

Dhurandhar, A. *et al.* (2018). Explanations based on the Missing: Towards Contrastive Explanations with Pertinent Negatives. In *Advances in Neural Information Processing Systems*, pages 592–603.

Fischer, F. *et al.* (2004). Hierarchical Reinforcement Learning in Communication-Mediated Multiagent Coordination. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems-Volume 3*, pages 1334–1335.

Gray, V. E. *et al.* (2017). Analysis of Large-Scale Mutagenesis Data To Assess the Impact of Single Amino Acid Substitutions. *Genetics*, **207**(1), 53–61.

Hendricks, L. A. *et al.* (2016). Generating Visual Explanations. In *European Conference on Computer Vision*, pages 3–19. Springer.

Iqbal, S. and Sha, F. (2019). Actor-Attention-Critic for Multi-Agent Reinforcement Learning. In *International Conference on Machine Learning*, pages 2961–2970. PMLR.

Jiang, M. *et al.* (2020). Drug–target affinity prediction using graph neural network and contact maps. *RSC Advances*, **10**(35), 20701–20712.

Jorgensen, W. L. *et al.* (1983). Comparison of simple potential functions for simulating liquid water. *The Journal of Chemical Physics*, **79**(2), 926–935.

Konda, V. R. and Borkar, V. S. (1999). Actor-Critic–Type Learning Algorithms for Markov Decision Processes. *SIAM Journal on control and Optimization*, **38**(1), 94–123.

Li, J. *et al.* (2016). Understanding Neural Networks through Representation Erasure. *arXiv preprint arXiv:1612.08220*.

Littman, M. L. (1994). Markov games as a framework for multi-agent reinforcement learning. In *Machine Learning Proceedings 1994*, pages 157–163. Elsevier.

McCloskey, K. *et al.* (2019). Using attribution to decode binding mechanism in neural network models for chemistry. *Proceedings of the National Academy of Sciences*, **116**(24), 11624–11629.

Meng, E. C. *et al.* (1992). Automated docking with grid-based energy evaluation. *Journal of Computational Chemistry*, **13**(4), 505–524.

Morimura, T. *et al.* (2009). A generalized natural actor-critic algorithm. *Advances in Neural Information Processing Systems*, **22**, 1312–1320.

Mothilal, R. K. *et al.* (2020). Explaining Machine Learning Classifiers through Diverse Counterfactual Explanations. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 607–617.

Nguyen, T. *et al.* (2020). GraphDTA: Predicting drug–target binding affinity with graph neural networks. *Bioinformatics*.

Numeroso, D. and Bacciu, D. (2020). Explaining deep graph networks with molecular counterfactuals. *Advances in Neural Information Processing Systems, Workshop on Machine Learning for Molecules*.

Öztürk, H. *et al.* (2018). DeepDTA: deep drug–target binding affinity prediction. *Bioinformatics*, **34**(17), i821–i829.

Öztürk, H. *et al.* (2019). WideDTA: prediction of drug-target binding affinity. *arXiv preprint arXiv:1902.04166*.

Perez-Liebana, D. *et al.* (2019). The Multi-Agent Reinforcement Learning in MalmÖ (MARLÖ) Competition. *arXiv preprint arXiv:1901.08129*.

Pope, P. E. *et al.* (2019). Explainability Methods for Graph Convolutional Neural Networks. In *"Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition"*, pages 10772–10781.

Preuer, K. *et al.* (2019). Interpretable Deep Learning in Drug Discovery. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, pages 331–345. Springer.

Pullman, A. (2013). *Intermolecular Forces*, volume 14. Springer Science & Business Media.

Raha, K. *et al.* (2007). The role of quantum mechanics in structure-based drug design. *Drug Discovery Today*, **12**(17-18), 725–731.

Rodríguez-Pérez, R. and Bajorath, J. (2019). Interpretation of compound activity predictions from complex machine learning models using local

approximations and shapley values. *Journal of Medicinal Chemistry*, **63**(16), 8761–8777.

Ryu, S. *et al.* (2018). Deeply learning molecular structure-property relationships using attention-and gate-augmented graph convolutional network. *arXiv preprint arXiv:1805.10988*.

Shang, C. *et al.* (2018). Edge Attention-based Multi-Relational Graph Convolutional Networks. *arXiv e-prints*, pages arXiv–1802.

Sutton, R. S. and Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

Tan, M. (1993). Multi-Agent Reinforcement Learning: Independent vs. Cooperative Agents. In *Proceedings of the Tenth International Conference on Machine Learning*, pages 330–337.

Thafar, M. *et al.* (2019). Comparison Study of Computational Prediction Tools for Drug-Target Binding Affinities. *Frontiers in Chemistry*, **7**.

Torng, W. and Altman, R. B. (2019). Graph Convolutional Neural Networks for Predicting Drug-Target Interactions. *Journal of Chemical Information and Modeling*, **59**(10), 4131–4149.

Tri, N. *et al.* (2020). GEFA: Early Fusion Approach in Drug-Target Affinity Prediction. *Advances in Neural Information Processing Systems, Workshop on Machine Learning for Structural Biology*.

Tsang, M. *et al.* (2018). Neural Interaction Transparency (NIT): Disentangling Learned Interactions for Improved Interpretability. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 5804–5813. Curran Associates, Inc.

Vaswani, A. *et al.* (2017). Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Veličković, P. *et al.* (2018). Graph Attention Networks. *International Conference on Learning Representations*.

Vermeire, T. and Martens, D. (2020). Explainable Image Classification with Evidence Counterfactual. *arXiv preprint arXiv:2004.07511*.

Wachter, S. *et al.* (2017). Counterfactual explanations without opening the black box: Automated decisions and the GDPR. *Harvard Journal of Law & Technology*, **31**, 841.

Ying, R. *et al.* (2019). GNNExplainer: Generating Explanations for Graph Neural Networks. *Advances in Neural Information Processing Systems*, **32**, 9240.

Zheng, S. *et al.* (2020). Predicting drug–protein interaction using quasi-visual question answering system. *Nature Machine Intelligence*, **2**(2), 134–140.

Zhou, J. and Troyanskaya, O. G. (2015). Predicting effects of noncoding variants with deep learning–based sequence model. *Nature Methods*, **12**(10), 931–934.

Zhou, Z. *et al.* (2019). Optimization of Molecules via Deep Reinforcement Learning. *Scientific Reports*, **9**(1), 1–10.