Md.Sadek Hossain Asif
Notre Dame College Dhaka
Bangladesh
asifsadek509@gmail.com

Using Deep Reinforcement Learning in Anti-Viral Cure of Covid (Orthocoronavirinae)

Abstract :

Predicting a drug through machine learning for a virus makes our work easy. Corona Virus is suddenly becoming a global threat. A potential cure for the virus is to have an inhibitor (Antibody) that will attach itself to the virus and prevent it from attaching to the human cell's receptors which will prevent it from spreading. In this paper, we first collected virus RNA sequences from the GISAID database, translated the RNA sequences into protein sequences. Then our model will give rank the drug according to the bond affinity with RNA of covid 19 and it is the model prediction. The lowest value of rank will have the highest bond affinity.Then we will collect 10 good samples for the candidate drug and from then any drug may be the anti-cure drug for covid. This series of sample candidate drugs will be able to detach covid19 from our cell receptor .

KeyWords: Deep Reinforcement Learning , Corona Virus , GAN (Generative Adversarial Network).

1. Introduction:

The coronavirus was first discovered in the mid-1960s [1], it is a large family of viruses and it causes illness that ranges from common cold to more severe illness like MERS-CoV, SARS-CoV. Recently, in Wuhan, China [2]. A new strain of the virus was discovered called the Novel Coronavirus or COVID-19. (CO – Coronavirus, VI – Coronavirus, D – December, 19 – 2019)
The coronavirus affects the human body by first entering the human cell through the Angiotensin Converting Enzyme 2 Receptors on the human cell membrane. By endocytosis, the human cell ingests the virus into its cytoplasm and the virus's genetic material which is a single-stranded RNA is revealed by the opening of its endosome. The virus replicates the RNA in n proteins of the human cell and through the endoplasmic reticulum, the virus forms the M protein outer layer and the S protein. Through exocytosis, the virus is carried out of the human cell by the Golgi bodies to affect other cells. This rate of viral production on the endoplasmic reticulum will lead the apoptosis of the human cell. There are currently several attempts to create an anti-viral drug to combat the virus. A research that claims that **Chloroquine** and **remdesivir** (which is an HIV inhibitor) has been recognized as effective candidates to control the virus [5]. Innovation Pharmaceuticals is evaluating **Brilacidin** as a candidate treatment for the virus, **CytoDyn** are also examining a potential cure for the virus called **leronlimab** [6]. Many other research facilities are also burning the mid-night candle to develop a potential treatment for the virus.

In this study we will predict a specific drug that will activate an antibody in our body which will prevent corona virus to attach with our body .Our model with text-classifier will match the pretrain dataset[10](chemical ligands samples) and the RNA of the covid 19 and then it will rank then according to bond affinity

Methods :

2.1 Requirements :

Tensorflow ,future ,editdistance ,PyMOL -Used to bind the virus to the 10 goodsample molecules

PyRX - Used to view structures of the virus e.g

2.2 Dataset :

For training the RNN model, we compiled a dataset of 677,044 SMILES strings with annotated nanomolar activities ($K_{d/i/B}$, IC/EC_{50}) from ChEMBL22 (www.ebi.ac.uk/chembl). The dataset was then pre-processed to remove duplicates, salts and stereochemical information. In addition, pre-processing filtered out nucleic acids and long peptides which lay outside of the chemical space from which we sought to sample. The RNN was ultimately trained on 541,555 SMILES strings, with lengths from 34 to 74 SMILES characters (tokens).For using the data perfectly we took help from[10].

2.3Model :

The deep learning model that will be used for this experiment is ORGAN which is a modified version of GAN (Generative Adversarial Network). The GAN model has two neural networks, which are Generator and Discriminator.[7]

The **Generator's** main objective is to generate fake samples that so closely resemble the true data/distribution that the discriminator can't distinguish between the true and the fake data[7] . As for the **Discriminator**, it discriminates the input data and classifies whether it is forming the true data sample/distribution, or is a fake sample generated by the Generator. It is initially trained on true samples of the labelled data[7] . Both networks are working against each other to try their work to prove themselves better and their main goal is to generate data points like some of the data points consisting if the training data. [7]

The Generator **G** samples x from a distribution $\mathbf{P_{synth}}$ (Fake Data), generated with random noise **z**, while the discriminator **D** looks at samples, either from $\mathbf{P_{synth}}$ (Fake Data) or $\mathbf{P_{data}}$ (Real Data) and attempts to classify their identity y as either real or fake[7]

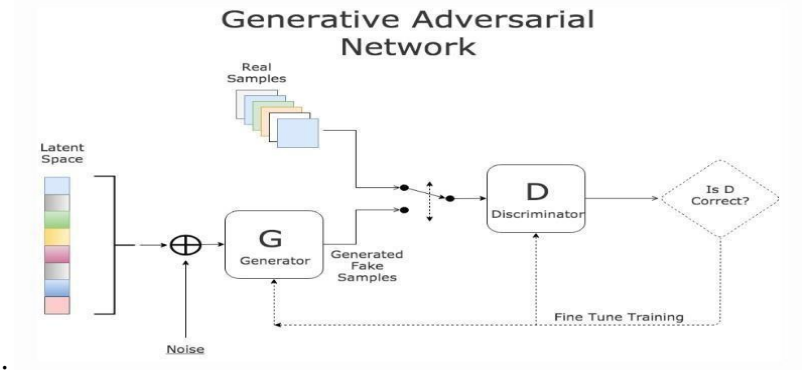

Figure 2: Generative Adversarial Network Structure [7]

According to [7], GAN's power and stability are limited by various problems. Whilst training the GAN, you will realize that the discriminator part is more powerful than its generator counterpart. The generator would thus fail to effectively train which will turn result in a huge loss in your GAN training process. But if the discriminator is too lenient, it would literally permit the generation of any image. This entire idea will thus remain useless to your GAN. To solve these problems according to [7], incorporating a Reinforcement learning to train the generator so that it will generate the desired output by updating the generator parameters with a policy gradient.

2.4 Reinforcement Learning Structure

According to [8], We treat the Generator **G** as an agent in an RL game with a policy gradient in which we consider states s, actions a from an action space **A**, and a reward function **Q**. A state $\mathbf{s_t}$ is a partial sequence of characters $\mathbf{X_{1:t}}$ which is already generated. We have an action space **A** that includes all possible characters to choose from for the next x_{t+1} character. Next, a reward function **Q(s,a)** which represents the expected reward for taking action **a** in state **s**. Each episode is the completion of a fully generated fixed length **T** sequence which is rewarded with the $\mathbf{R_T}$ **(X)** function. The stochastic policy of the agent is given by $(\boldsymbol{y_t|Y_{1:t-1}})$and we want to maximize the long-term reward **J(w)** expected:

$$\boldsymbol{J(\theta)}=\mathbf{E}[\boldsymbol{RT|s\theta,\theta}]=\sum_{\mathbf{x1}\in\mathbf{X}} \mathbf{G\theta(x1|s\theta)}.\ \mathbf{Q(s\theta, x1)}$$

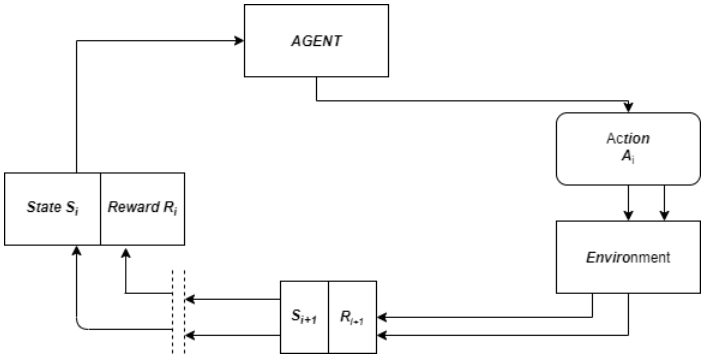

Figure 3: Reinforcement Learning Structure [7]

It should be noted as stated by [7] that molecular metrics such as Drug-likeness, Natural Product Likeliness were used to maximize the reward by optimizing the generator to generate similar molecules to the initial data distribution. The discriminator then analyses the generated molecules and the reward metric. It then optimizes the generator, and trains it to fool the discriminator.

The above steps where the **Pre-Training** phase which constituted the first half of the training. For the final half of the training, policy gradient is used to train both the generator and the discriminator. The loss is calculated for each character generated, and the model updated. The policy gradient loss is calculated in the case of Generator. Afterwards the generator is optimized, and all parameters are updated [7]. The policy function calculates the output sequence Log SoftMax given rewards, the destinations and the sequence length. Its output is negative because we want to minimize loss but maximize the gradients of policy. [7]

$$\boldsymbol{L = -Q(s,a)\log\left(G(y_t|Y_{1:t-1})\right)}$$

Where **Q(s,a)** expected reward for an action **a** in state **s** and $\boldsymbol{G(y_t|Y_{1:t-1})}$.

3.Experiments and Results

The experiment performed in this research was greatly influenced by the efforts of [9]. Firstly, the Deep Reinforcement Learning model (ORGAN) discussed above was used to generate potential candidates' drugs (**Drug Discovery**). The data set or SMILES used for this experiment were gotten from [10]. According to [11], the SMILES should cleaned to remove salts, duplicates and stereochemical information. The molecular matrix or the objective was set to **Solubility** which indicates how likely a molecule can mix with water. The SMILES were put into the model and trained. After about 9 hours of training, with a λ of 0.2 and epochs of 240 and a sample set of 6400,

10 good sample SMILES were generated and The **Solubility or LogP** of these samples is **0.7098.** see Table 1:

One of the Good Smiles look Like :

Cc1ccc(F)cc1C(C)c1cccc(C(=O)c2ccccc2)n1
Cc1ccc(Cl)cc1OC(=O)c1ccc(C)cc1-c1cnnnc1N
COc1ccc(C(=O)Nc2ccc(Br)cc2)c2cc(Cl)ccc12
CCNCC=Cc1cc(C(=O)NOCc2ccccc2)cc(Cl)c1OCC
CC(=C)C(=O)c1ccc(C(=O)c2ccccc2)cc1CCS
COc1ccc2c(c1)c(Nc1cccc1)c1cnc2cccc(C)c1
COc1cccc(C(=O)Nc2ccnc(OCCCc3ccccc3)n2)c1
O=C(O)CC(c1ccccc1)c1cccc(-c2ccccc2)c1
Oc1nc(SCc2c[nH]c3nc[nH]c3s2)cc2sccc12
N#Cc1ccc(CCCCCCCCOc2ccc3nc(N)ncn32)cc1Cl

Other molecular matrixes for the 10 good samples were also recorded as displayed in the table 1. The **novelty** of the sample drugs which shows how molecularly distinct these samples drugs are from the other candidate drugs is seen to have a high value of 1.The **synthesizability** which shows or indicates how hard (0) or easy (1) it is to synthesize these 10 good sample molecules has the value of 0.6197.

Novelty	1.0000
Hard Novelty	1.0000
Soft Novelty	1.0000
Diversity	0.4698
Conciseness	0.9866
Solubility	0.7098
Naturalness	0.5828
Synthesizability	0.6197

Table 1: Molecular Matrixes

After these 10 good sample candidates' drugs were collected, there were all bonded to the corona virus through the **PyRX software**. The binding affinities of each of the samples were recorded with the results displayed the figure below.

	Ligand	Binding A	rmsd/ub	rmsd/lb
2	6lu7_C18H15ClN4O2_uff_E=471.56	-7.3	0	0
3	6lu7_C23H20N2O_uff_E=2008.31	-6.9	0	0
4	6lu7_C21H18FNO_uff_E=469.56	-6.8	0	0
5	6lu7_C18H15ClN4O2_uff_E=471.56	-6.8	6.651	3.671
6	6lu7_C21H18FNO_uff_E=469.56	-6.5	7.139	2.851
7	6lu7_C21H18O2_uff_E=353.61	-6.4	0	0
8	6lu7_C21H18FNO_uff_E=469.56	-6.3	20.923	18.893
9	6lu7_C21H18FNO_uff_E=469.56	-6.2	0	0
10	6lu7_C18H15ClN4O2_uff_E=471.56	-6.2	6.017	3.844

The lowest value has highest bond affinity

Figure 4: Binding Affinities of candidate drugs

From figure 4, the drug with the **highest binding affinity** (Most negative value) is **C18H15ClN4O2** also known as **Olutasidenib**. This is **best candidate drug** that has a potential of being an anti-viral cure to the corona virus. According to [12] , **Olutasidenib** is an active, selective mutant Isocitrate dehydrogenase (IDH)1 inhibitor for treating acute myeloid leukemia. The results produced through the experiments conducted above tends to convey that this drug could also be a potent cure for the corona virus.

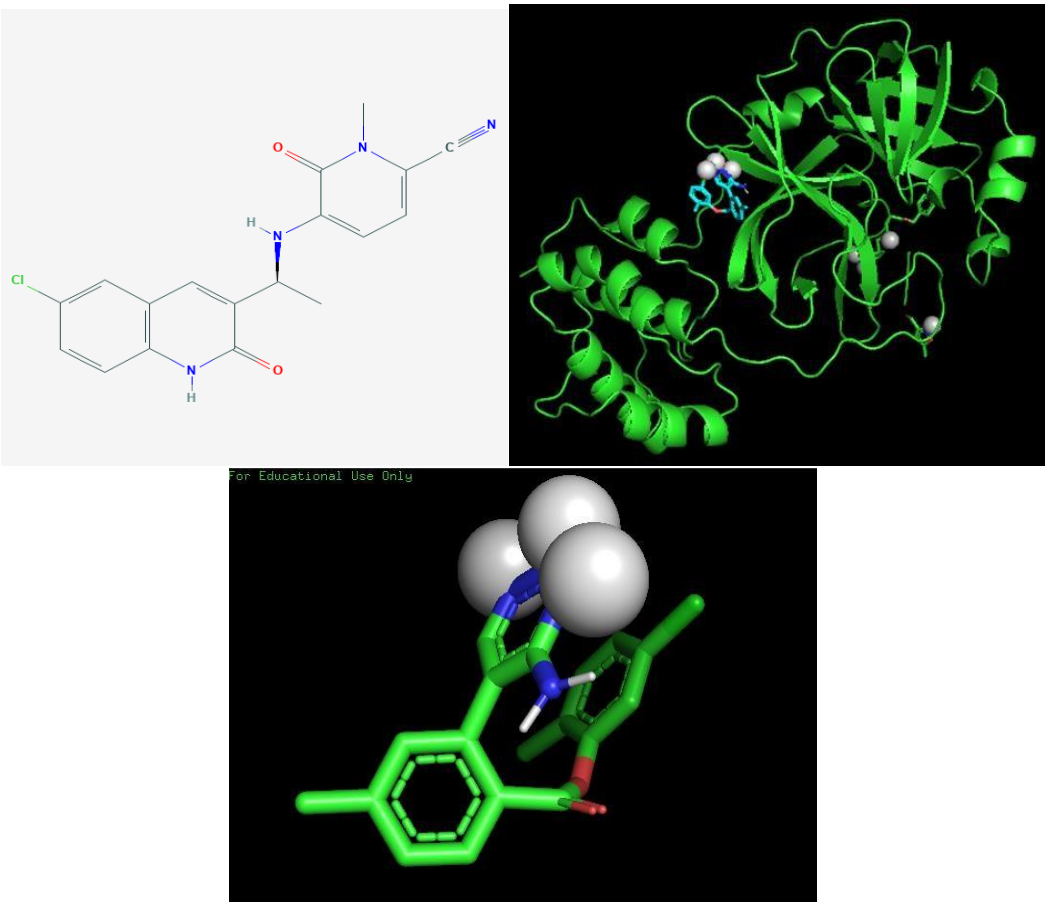

*Figure 5: Top left: 2d structure of the **Olutasidenib** drug. Top right: Using pyMOL, image of the **Olutasidenib** been bounded the corona virus. Bottom: Using pyMOL, 3d structure of the **Olutasidenib** drug.*

3 Conclusion:

The corona virus is rapidly increasing with a global death toll of over 50,000 as of the time of this experiment, this value is forecasted to increase. The deep reinforcement learning model used in this experiment/research (ORGAN) was able to identify 10 sample molecules or LINGADS which were then bounded to the virus and the candidate molecule was found. Further work should be approved by taking the candidate drug (**Olutasidenib)** to the lab to be synthesized andtested.

4 Reference

[1] "Coronavirus | Human Coronavirus Types | CDC." [Online]. Available: https://www.cdc.gov/coronavirus/types.html. [Accessed: 25-Feb-2020].

[2] "Coronavirus." [Online]. Available: https://www.who.int/health-topics/coronavirus.

[Accessed: 25-Feb-2020].

[3] "Coronavirus - an overview | ScienceDirect Topics." [Online]. Available: https://www.sciencedirect.com/topics/neuroscience/coronavirus. [Accessed: 28-Feb-2020].

[4] "(770) Medical Animation explaining Coronavirus MOA - How Coronavirus attacks a human body (UPDATED) - YouTube." [Online]. Available: https://www.youtube.com/watch?v=5GELx45kWPw. [Accessed: 25-Feb-2020].

[5] M. Wang *et al.*, "Remdesivir and chloroquine effectively inhibit the recently emerged novel coronavirus (2019-nCoV) in vitro," *Cell Research*. Springer Nature, pp. 1–3, 04-Feb-2020.

[6] "Coronavirus outbreak: Top coronavirus drugs and vaccines in development." [Online]. Available: https://www.clinicaltrialsarena.com/analysis/coronavirus-mers-cov-drugs/. [Accessed: 27-Feb-2020].

[7] "Objective Reinforced Generative Adversarial Network (Part I) | BayesLabs blog." [Online]. Available: https://blog.bayeslabs.co/2019/07/03/Objective-Reinforced-Generative-Adversarial-Network-(Part-I).html. [Accessed: 27-Feb-2020].

[8] B. Sanchez-Lengeling, C. Outeiral, G. L. Guimaraes, and A. Aspuru-Guzik, "Optimizing distributions over molecular space. An Objective-Reinforced Generative Adversarial Network for Inverse-design Chemistry (ORGANIC)," *ChemRxiv*, pp. 1–18, Aug. 2017.

[9] M. Benhenda, "ChemGAN challenge for drug discovery: can AI reproduce natural chemical diversity?," 2017.

[10] "GitHub - topazape/LSTM_Chem: Implementation of the paper - Generative Recurrent Networks for De Novo Drug Design." [Online]. Available: https://github.com/topazape/LSTM_Chem. [Accessed: 27-Feb-2020].

[11] A. Gupta, A. T. Müller, B. J. H. Huisman, J. A. Fuchs, P. Schneider, and G. Schneider, "Generative Recurrent Networks for De Novo Drug Design," *Mol. Inform.*, vol. 37, no. 1, Jan. 2018.

[12] C. Wang *et al.*, "Inducing and exploiting vulnerabilities for the treatment of liver cancer," *Nature*, vol. 574, no. 7777. Nature Publishing Group, pp. 268–272, 10-Oct-2019.