Relatório: *Protein-Ligand Scoring with Convolutional Neural Networks* - Avaliação da Tarefa de Virtual Screening

Guilherme R. Graeff [1]

[1] *Structural Bioinformatics and Computational Biology Lab - SBCB, Universidade Federal do Rio Grande do Sul, Porto Alegre, RS, Brasil.*

Correspondence*:
Guilherme Rafael Graeff
guilherme.graeff@ufrgs.br

1 INTRODUÇÃO

Este trabalho faz parte da avaliação da disciplina de Algoritmos Para A Bioinformática E Biologia Computacional. Compreende uma apresentação sobre uma aplicação de Redes Neurais Convolucionais para a tarefa de *Virtual Screening* [8]. Explorando a abordagem utilizada para a representação dos dados de estrutura molecular, analisando os *datasets* escolhidos para o treinamento do modelo e verificando os resultados obtidos pelo trabalho.

O restante do texto está organizado da seguinte forma: a próxima seção apresenta a área de Aprendizado de Máquina e *Docking* Molecular. A descrição dos dados utilizados esta presente na Seção 3. A seção 4 está dedicada a apresentação da aplicação. Em seguida, são discutidos os resultados do trabalho na seção 5. Enfim, a seção 6 conclui o relatório.

2 CONTEXTUALIZAÇÃO

Esta seção apresenta conceitos presentes no trabalho, acompanhado de uma breve explicação importante para a compreensão da técnica utilizada.

2.1 Aprendizado de Máquina - Redes Neurais Convolucionais

Rede Neural Artificial [3] é uma técnica de Aprendizado de Máquina capaz de compreender padrões presentes em determinado conjunto de dados, possui conceitos inspirados em neurônios biológicos, possui também a capacidade de aprender a partir de uma função de erro. O objeto de estudo do trabalho minimiza a perda logística multinomial utilizando uma variante da descida gradiente estocástica (SGD) e *backpropagation* para o treinamento. Uma Rede Neural Convolucional (CNN) [6] é uma rede neural que possui camadas de convolução e camadas de *pooling*, esta técnica é amplamente utilizada para o reconhecimento de imagens no campo da Visão Computacional[6].

A convolução consegue captar informação conformacional do dado, como por exemplo arestas de objetos em uma imagem, o mesmo se aplica quando o dado possui mais dimensões. Já a camada de *pooling* é capaz de reduzir a dimensionalidade do dado para que operações sejam realizadas neste 'menor' espaço, ou seja, esta seria a entrada para a rede neural totalmente conectada no fim da arquitetura da rede. A convolução consegue capturar as características que definem o modelo, então não é necessária a extração de *features* relevantes do mesmo.

2.2 *Docking* molecular com Smina

A ferramenta utilizada para *Docking* molecular foi o *Smina*[5], uma implementação derivada do *Autodock Vina* [12], este algoritmo que a partir dos dados estruturais do alvo e do ligante retorna as melhores poses para o ligante. Neste contexto, dois conceitos são importantes para a identificação de um exemplo positivo e um exemplo negativo, são chamadas *decoy* aqueles que são exemplos de de controle negativos pois são moléculas que não possuem interagem com o receptor, ao contrário das moléculas ativas que possuem afinidade e interagem com o receptor. Um 'Alvo' por sua vez é a molécula receptora do ligante, esta fica estática durante o processo de *docking*.

3 CARACTERÍSTICAS E PREPARAÇÃO DOS CONJUNTOS DE DADOS

Há mais de um conjunto de dados utilizado no trabalho, por conta das diferentes tarefas que se busca realizar, este trabalho apenas explica os dados utilizados para a tarefa de *Virtual Screening.*

3.1 Dados de treinamento

A tarefa de *Virtual Screening* faz uso do Databese of Usefull Decoys - Enhenced (DUD-E)[7]. São 102 alvos(proteínas), 20000 moléculas ativas e mais de um milhão de moléculas do tipo *decoy*. Este banco de dados não possui a cristalografia da pose dos ligantes, embora possua uma referência do complexo disponível.

3.1.1 Gerando poses para treinamento

São geradas poses de ligantes para moléculas ativas e *decoy* utilizando *Docing with Smina*, *Smina* utiliza a função de score do *Autodock Vina* [5]. A molécula é posicionada na posição de referência do alvo, e o *docking* acontece em uma caixa que possui 8Å centrada à este ligante de referência. Caso não exista um ligante de referência, então é utilizado um *script* que define a posição da caixa.

Os ligantes são atracados com a referência utilizando os argumentos padrão do *smina* para os parâmetros *exhaustiveness* e *sampling*. Então é selecionada a pose com o melhor ranking definido pela função de score do *Autodock Vina*. O tamanho final dos dados de treinamento são de 22.645 exemplos positivos e 1.407.141 exemplos negativos. O valor expressivo de exemplos negativos se dá melo maior número de *decoy* presente no conjunto de dados.

3.2 Conjuntos de testes independentes

O trabalho opta por avaliar a acurácia da classificação com um conjunto de dados de teste independente, garantindo que dados utilizados no treinamento não estejam presentes no teste. Dos dois conjuntos de dados escolhidos para o teste da tarefa de *Virtual Screening*, um foi gerado através do *ChEMBL* por Riniker e Landrum [9], seguidos por Heikamp e Bajorath [4]. O outro conjunto de dados é um subconjunto dos dados presentes em *maximum unbiased validation (MUV) dataset*[10] que é baseado nos dados de bioatividade presentes no PubChem.

Estes conjuntos de testes independentes ainda são filtrados, antes que o modelo seja testado. Por exemplo, dentre outras técnicas, uma delas remove quaisquer alvos que possuam 80% ou mais de identidade de sequência com um alvo de treinamento. Destes dados, apenas fazem parte do conjunto final de testes aqueles que possuem o complexo de ligação contendo o alvo disponível no *Protein Data Bank*[1] . Estas estruturas são utilizadas para gerar poses atracadas em um sítio de ligação conhecido.

Após a curagem dos dados, os conjunto de testes independentes para a tarefa de *Virtual Screening* consiste de 13 alvos provenientes de Riniker e Landrum ChEMBL [9], e 9 alvos provenientes do conjunto MUV[10].

4 REFLEXÕES SOBRE A APLICAÇÃO

4.1 Entendendo a utilização de um Grid

A biblioteca utilizada para realizar a transformação dos dados em um *grid* possui ligações em *Python* através do pacote de código aberto *libmolgrid* [11]. Este *grid* é um *array* multidimensional, este *array* prove uma distribuição contínua da entrada.

4.2 Utilizando rede neural convolutiva

Utilizar Funções de Score baseadas em Redes Neurais Convolucionais traz uma maneira abrangente de representar a estrutura tridimensional de uma interação proteína e ligante. A técnica em questão utiliza um *grid* de densidade de átomos [8] gerados a partir da estrutura, a biblioteca *libmolgrid* [11] é responsável pela transformação dos dados. Este tipo de modelo consegue aprender as principais características, relativas ao atracamento, da interação entre proteína e ligante [8]. Para a tarefa de *Virtual Screening* é treinada e otimizada para diferenciar ligantes e não-ligantes conhecidos. A técnica abordada no trabalho demonstra competitividade em relação a outras funções de *scoring*. O modelo foi otimizado utilizando a técnica de *clustered cross-validation*

5 DISCUSSÃO DE RESULTADOS

Ao avaliar a tarefa de *Virtual Screening* são considerados dois casos, o primeiro leva em consideração apenas a pose que está no topo do ranking de poses atracadas utilizando o *Vina*[5] (*single-pose prediction*). No segundo dos casos o modelo seleciona de todas as posições de atracamento disponíveis do ligante(multi-pose prediction). O modelo utiliza a a seguinte métrica, área de baixo da curva (AUC) *Receiveroperating characteristic* (ROC), AUC = 1 representa um classificador perfeito e AUX = 0.5 indica que o modelo não é melhor do que a escolha soluções aleatórias.

5.1 Utilizando dados de treinamento

É feita uma análise isolada, apenas com os dados do DUD-E[7], resultando em CNN *scoring* superar o Vina com AUC de 0.85 contra 0.68 porém dependente deste *dataset*, modelo não tão generalista. Também é realizada uma análise que combina os conjuntos de dados utilizados no treinamento do modelo referente a tarefa de predição de pose. Estes testes evidenciam que há diferença ao utilizar um modelo para classificar outro *dataset*. A combinação dos dados para o treinamento de um modelo combinado, com o fim de o modelo se tornar mais generalista, utiliza a proporção 2:1 de dados presentes no conjunto DUD-E e CSAR[2]. A versão que usa os dados combinados atinge uma AUC de 0.83. Amostras *outliers* se destacaram por ser muito bem avaliada quando utilizados apenas dados do DUD-E e em contra partida de um se sair mal avaliada ao ser classificada pelo modelo que utilizou os dados combinados.

5.2 Utilizando conjuntos de testes independentes

Com os modelos treinados então são utilizados ambos DUD-E quanto a combinação dele com o CSAR para analisar os resultados. O modelo é então é avaliado nos conjuntos de testes ChEMBL e MUV. Estes *datasets* são mais desafiadores, para o ChEMBL os resultados são: Vina 0.67, 0.64 e 0.78, DUD-E CNN

0.71, 0.80 e 0.86. Para o MUV: Vina 0.55, 0.50 e 0.52. O modelo traz a reflexão sobre a influência da construção e escolha do conjunto de dados, neste caso específico por conta da diferença na avaliação dos modelos, pode se concluir que a construção do dado pode influenciar, então diferentes abordagens para a construção dos *decoys* podem trazer diferentes resultados.

6 CONCLUSÃO

Este trabalho apresenta o tema de Redes Neurais Convolucionais utilizadas em biologia estrutural, uma área multidisciplinar que envolve profundo conhecimento tanto sobre computação quanto sobre biologia. A ideia inicial do trabalho seria realizar a reprodução do artigo selecionado, a possibilidade de reprodução não obtiveram sucesso ao se deparar com a complexidade referente a esta tarefa. Então o trabalho tomou uma forma que busca contextualizar os colegas quanto a utilização de CNN em dados que possuam forma que permita a sua utilização, estudando um pouco o formato de entrada utilizado no trabalho de *M. Ragoza* [8]. O relatório é uma acaba se tornando uma ferramenta de reflexão, não só sobre o artigo objeto de estudo, mas também sobre o processo de aprendizagem, se referindo a dificuldade na realização do trabalho, sobre a mudança de planos, e sobre a apresentação e compreensão do conteúdo.

REFERENCES

[1]H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, I.N. Shindyalov, and P.E. Bourne. The protein data bank. *Nucleic Acids Research*, 28(1):235–242, 2000.

[2]J. B. Dunbar, R. D. Smith, C.-Y. Yang, P. M.-U. Ung, K. W. Lexa, N. A. Khazanov, J. A. Stuckey, S. Wang, and H. A. Carlson. Csar benchmark exercise of 2010: Selection of the protein-ligand complexes. *Journal of Chemical Information and Modeling*, 51(9):2036–2046, 2011.

[3]J. Feldman and R. Rojas. *Neural Networks: A Systematic Introduction*. Springer Berlin Heidelberg, 2013.

[4]K. Heikamp and J. Bajorath. Large-scale similarity search profiling of chembl compound data sets. *Journal of Chemical Information and Modeling*, 51(8):1831–1839, 2011. Epub 2011 Jul 14.

[5]David Ryan Koes, Matthew P Baumgartner, and Carlos J Camacho. Lessons learned in empirical scoring with smina from the csar 2011 benchmarking exercise. *Journal of chemical information and modeling*, 53(8):1893—1904, August 2013.

[6]Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436, 2015.

[7]Michael M. Mysinger, Marco Carchia, John J. Irwin, and Brian K. Shoichet. Directory of Useful Decoys, Enhanced (DUD•E): Better Ligands and Decoys for Better Benchmarking. *Journal of Medicinal Chemistry*, July 2012.

[8]Matthew Ragoza, Joshua Hochuli, Elisa Idrobo, Jocelyn Sunseri, and David Ryan Koes. Protein–ligand scoring with convolutional neural networks. *Journal of Chemical Information and Modeling*, 57(4):942–957, 2017. PMID: 28368587.

[9]S. Riniker and G. A. Landrum. Open-source platform to benchmark fingerprints for ligand-based virtual screening. *Journal of Cheminformatics*, 5:1–17, 2013.

[10]S. G. Rohrer and K. Baumann. Maximum unbiased validation (muv) data sets for virtual screening based on pubchem bioactivity data. *Journal of Chemical Information and Modeling*, 49:169–184, 2009.

[11]Jocelyn Sunseri and David Ryan Koes. libmolgrid: Gpu accelerated molecular gridding for deep learning applications, 2019.

139 [12]Oleg Trott and Arthur J. Olson. Autodock vina: Improving the speed and accuracy of docking with a
140 new scoring function, efficient optimization and multithreading. *Journal of Computational Chemistry*,
141 31(2):455–461, 2010.