To Block or Not to Block: Predicting hERG inhibition using ML Methods

Jeffrey T. Campbell, Tushar Agashe, Somayeh Motevalli and Shiyang Xin

AICDDD, University of California San Fransisco, SF, CA, USA

My group chose the hERG inhibition data set by Karim et al. due to the medical relevance of predicting hERG blocking, a critical factor in assessing the safety of drug candidates. Predicting hERG inhibition is essential for identifying compounds that could potentially cause cardiac arrhythmias, making this a critical problem for both pharmaceutical development and patient safety. Our objective was to create a model that could accurately predict hERG inhibition from the Simplified Molecular Input Line Entry System (SMILES) provided by the Karim et al. paper.  As a group, we approached this problem by training multiple different ML models on the data set and comparing their performance to see which would perform the best. In my individual approach I tried to pick an ML method that would best capture the structural information of the molecules in the data set, as well as information about the spacial relationship of functional groups, which plays a large role in all biological activity, including hERG inhibition. For this reason I chose to use a GNN that would be trained on molecular graphs created from the SMILES strings.

# Introduction

﻿ The hERG (human ether-a-go-go-related gene) channel is a voltage-gated potassium channel critical for cardiac repolarization, specifically during the phase 3 of the cardiac action potential. It plays a pivotal role in maintaining normal heart rhythm by ensuring proper timing of cardiac electrical activity. Blocking hERG channels is essential in drug development because unintended inhibition of this channel can prolong the QT interval on an electrocardiogram (ECG), potentially leading to life-threatening arrhythmias such as Torsades de Pointes. As a result, screening for hERG blockade has become a standard part of preclinical safety pharmacology to minimize cardiotoxicity risks in new drugs, while ensuring therapeutic compounds do not disrupt normal cardiac function.

The Data set we used to train our models, provided by Karim et al. contains the SMILES strings and a binary classification for blocking vs non-blocking of hERG for 2689 molecules. These have an average size of 40 atoms per molecule with a standard deviation of 6.16 atoms as can be seen in figure 1, meaning that it is almost exclusively small molecules, and would not be informative for prediction with large or organic molecules. Most atoms in these molecules are carbon organized into aromatic rings, with the ketones, amines, halogens, and alcohols representing almost all of the other functional groups as can be seen in figure 2. Another feature that would make this data set not applicable for prediction of hERG inhibition by natural compounds is the high prevalence of fluorine, an atom that is almost never found in natural products.

A diagram of a number of atoms

Description automatically generated

Figure 1. Distribution of molecular size in Karim et al. data set

A graph of a number of groups

Description automatically generated

Figure 2. Frequency of functional groups in Karim et al. data set

# Methods

The first issue our group faced was that SMILES are strings, and therefore could not be used directly as input to our ML models and so we needed to convert them to data our models could use. To do this, we utilized Morgan fingerprints, a common method for encoding molecular structures into high-dimensional numeric vectors. Morgan fingerprints encode information about a molecule's structure by iteratively identifying and hashing circular substructures (like atom neighborhoods) into fixed-length binary or numeric arrays. This representation captures the molecule's structural features and relationships in a compact, high-dimensional format. Once we converted the SMILES into Morgan fingerprints, we also added MACCS Fingerprints which characterize common chemical structures with known properties  and Molecular Descriptors, which assign numerical values to macro level characteristics of a molecule such as Molecular weight, LogP, Topological polar surface area, and Hydrogen bond donors/acceptors. We used these features in addition to Morgan Fingerprints so we could capture more information about the molecules and their activity beyond just their structure. We also used RandomizedSearchCV to automatically optimize the tuning of our hyperparameters and used a standard 70, 10, 20 split for training, test, and validation datasets from the Karim data set.

For my individual approach I decided to encode the SMILES as molecular graphs and use them to train a Graphical Neural Network. I chose this approach because it subverts the very high dimensionality of Morgan fingerprinting which is typically fixed at 2048 bits, whereas molecular graphs have dimensionality that scales with the size of the molecule. I chose this approach because the molecules that are used in the data set are quite small, with an average size of about 40 atoms, and could be represented with molecular graphs with less than 300 dimensions. Molecular features can also be added to nodes in molecular graphs, allowing chemical information about the specific regions of the molecule to be considered, not just global features such as total polar surface area. I included 9 molecular features I believed would be predictive of hERG inhibiton:  Atomic number, Degree, Hybridization, Formal charge, Aromaticity, Explicit valence, Implicit valence, Atomic mass, and Ring membership.

# Results

The group's best model, XGBoost performed at 87 percent accuracy for positive and negative classification with an f1 score of 0.87 and a recall of 0.86. My GNN model, when allowed to train for 600 epochs, performed at very similar levels with an accuracy of 0.88, and F1 of 0.86 and a recall of 0.87. In terms of raw performance, these models have almost identical metrics, which may indicate that they are both approaching the limit of what can be learned from this data set. However I needed to run my GNN model for 600 epochs in order to obtain this level of performance which took approximately 10 minutes, whereas the XGboost model trained in a matter of seconds. Given their similar performance, it is clear to me that for this data set, the XGboost method is the more efficient. I believe that this is due to GXboost being much better optimized for larger data sets and taking advantage of parallel processing.

While my group used a function called RandomizedSearchCV to automatically estimate optimal weights, I did not. This was because my model could already get similar performance to XGboost without tuning. However, I believe that going back using this package or a similar one to optimize the weights of my features may reduce the time and computation needed to achieve similar performance. The two major forms of tuning that I did for my model were number of epochs and number of features considered. These efforts were largely effective as my starting model used only molecular weight and 10 epochs and produced a 0.5 accuracy, whereas my final model using 600 epochs and 9 features achieved 0.87 accuracy.

# Discussion

The major limitation to the approach I have used is that it is very computationally intensive to train compared to my group's method. To improve my model accuracy, I could select more and different features and adjust their importance before training. However, I do not believe that there is a way to significantly reduce the time taken to train the model to be on parity with the XGboost model. The XGboost is simply designed from the ground up to be faster on these types of data sets. Altering the weights and features, as well as number of epochs may allow my model to work faster, and perhaps even outperform XGboosts for accuracy as it better capures spacial relations, but it will not outperform in terms of train time.

# References

1. Karim A., Lee M., Balle T., Sattar A. CardioTox net: a robust predictor for hERG channel blockade based on deep learning meta-feature ensembles. J. Cheminf. 2021;13(1):1–13. doi: 10.1186/s13321-021-00541-z