A Minimal-Input Multilayer Perceptron for Predicting Drug-Drug Interactions Without Knowledge of Drug Structure

10.17975/sfj-2020-006

Alun Stokes¹, William Hum^{1,†}, Jonathan Zaslavsky^{1,†}¹McMaster University[†]These authors contributed equally to this work

ABSTRACT: Predictive models in the drug discovery industry have an essential role that cannot be understated. With the sheer volume of potentially useful compounds that are considered for use, it is becoming more difficult to investigate overlapping interactions between two drugs. Given that recreational drugs lack the rigorous warnings of prescription drugs, it is important for the layperson to know which drugs can and cannot mix. Other methods are necessary to bridge this knowledge gap in the absence of deterministic experimental results for every drug combination. Ideally, such methods would require minimal inputs, have high accuracy, and be computationally feasible. We have not encountered a model that meets all the above criteria. In light of this, we propose a minimal-input multi-layer perceptron that predicts the interactions between two drugs. This model has a great advantage of requiring no structural knowledge of the molecules in question, and instead only uses experimentally accessible chemical and physical properties; in particular, 20 per compound. These 20 were selected as they seemed they may be pertinent to the authors, although given this was an exploratory model, there was not a rigorous method behind the choice. Using a set of known drug-drug interactions and associated properties of the drugs involved, we trained our model on a dataset of about 650,000 entries. We report an accuracy of 0.968 on unseen samples of interactions between drugs on which the model was trained and an accuracy of 0.942 on unseen samples of interactions between unseen drugs. We believe this to be a promising and highly extensible model that has the potential for high generalized predictive accuracy with further tuning.

KEYWORDS: drug interaction, neural network, binary classification, minimal-input model, multi-layer perceptron

INTRODUCTION

As the drug discovery process advances along with the increasing prevalence of drugs in our lives, understanding somatic drug-drug interactions is a pressing problem. While delineating the effects of given individual drugs in the body is important, it is also necessary to identify how it interacts with other compounds. Exhaustively testing every drug-drug combination is both cost and time-prohibitive due to the sheer volume of compounds to be tested, making optimized methods necessary to perform proper analysis [1]. A practical approach for identifying potential drug interactions and assessing associated risks has emerged in recent years with the development of predictive pharmacological networks [2]. Far-reaching applications of such methods with respect to the drug discovery process could provide easy dynamic tools for laypeople to better control their health. Correlated

to an increasing prevalence of recreational drug usage are many drug-related ailments that could have otherwise been prevented by not mixing mutually incompatible drugs. As such, developing accessible, fast models to determine these interactions is of significant societal importance [1]. In this paper, we propose the use of machine learning methods and known drug-drug interactions to predict these interactions in unknown drug pairings, as well as drugs on which a model has not been trained.

One of the largest barriers to predicting drug-drug interactions is the complexity of factors associated with that interaction. As such, many deterministic or exhaustive methods fail the test of feasibility due to their necessary computational complexity. Even such a problem as codifying the molecular structure of a chemical leads to a great complexity of information to be processed, with this processing being non-trivial of its own merit. Therefore,

any strategy to predict these interactions reasonably and at scale must do so on a minimal set of parameters, with limited computational complexity. Our model fits these requirements given each drug molecule is parameterized by 20 floating-point values and all of the necessary computation consists of matrix operations that are hardware-accelerated on modern computers. This leads to a model which, on an average consumer laptop processor (Intel® Core i5-8259U), can predict interactions for over 10,000 drug pairs per second – with approximately 0.968 accuracy. This model determines the binary condition of interactivity, rather than categorizing the type or location of interaction between two drugs. The implementation of the latter is, however, expected to progress naturally from the presented model, and would only require superficial adjustments. The reason for using a less specific model in this paper is two-fold: the limited accessibility of a dataset that characterizes the location and types of interactions as well as the amount of time allotted for completion of this project. The drug-drug interaction dataset used in this paper was provided by The Science and Engineering of Emerging Systems Lab [3], who collected their data from DrugBank. The molecular properties of each drug were obtained from PubChem [4], a database maintained by the National Centre for Biotechnology Information.

METHODS

The primary dataset used in this paper [3], is a line-separated compilation of lists of drugs that are known to interact with one another. There are approximately 1,200 unique species documented, and about 225,000 individual drug-drug interactions. A conservative estimate indicates that about 3.1% of these compounds are used recreationally. Certain parts of the dataset were unusable, as they specified macromolecule-drug interactions, and we are only interested in small-molecule drugs. Additionally, certain compounds did not have entries [4] that could be programmatically downloaded and processed, so those interactions were discarded. This left about 190,000 drug-drug interactions for the training, validation, and testing of the model. A larger and more generalized set would better help the model train to determine the important factors in a drug-drug interaction. As such, several data augmentation techniques were performed on the set in order to increase its size, whilst not perverting the accuracy or validity of its information.

First, given that we only had records of drug-drug interactions, we needed to generate a set of drug-drug non-interactions. Given that this dataset lists every subset of the drugs that do interact, permuting two drugs in the set, and checking for their inclusion in the original dataset could generate a list of non-interacting drugs [3]. This doubled the size of the original dataset – although it should be noted that doubling was chosen to keep the training categories evenly distributed. This improves training performance and avoids the necessity for weighting the training rates per category. Next, each 2-tuple of interacting drugs was reversed and added back into the list, again doubling the dataset. The order in which the two drugs are fed into the model should be independent of their interaction, and as such it should be trained to find an interaction between two drugs, regardless of which is first in the list. This brought the size of the dataset up to approximately 760,000 drug pairs. This set was then shuffled to prevent the model from learning any patterns present in the original presentation of the data. Of this full set, 10% is set aside for validation during training, and a further 10% composes the testing set. These are chosen randomly, although for the purposes of comparing models, the generator seed was fixed to allow comparable statistics.

Once the interaction set was processed, information of the individual compounds needed to be associated with each drug in an interaction. This data was obtained from Kim et al. [4] and certain attributes were extracted and assigned to the interacting molecules to be fed into the model. The full list of extracted properties can be seen in Table 1. Note that although these are mostly integer-valued parameters, for the purposes of computation they are treated as floating-point values. The neural network is a feed-forward network, meaning propagation only goes in the forward direction. Specifically, it is a multi-layer perceptron, with numerous hidden layers. Given the assumed non-linearity of this problem, at least several latent layers were necessary to predict interactions with any degree of accuracy. Each latent layer is either a fully-connected (FC) layer, a dropout layer, or a batch normalization (BN) layer. The network follows a diamond-like structure for the number of nodes per layer, starting with 256, increasing over several layers to 512, and then decreasing over several layers to 128 before the final output. Every FC layer is followed by a BN layer. Table 2 breaks down each layer of the network. Additionally, layers are grouped together to create a 'super-layer' structure

Table 1. Chemical parameters for input into model.

Parameter	Type
Number of C atoms	Integer
Number of H atoms	Integer
Number of N atoms	Integer
Number of O atoms	Integer
Number of P atoms	Integer
Number of S atoms	Integer
Number of F atoms	Integer
Number of Cl atoms	Integer
Number of I atoms	Integer
Number of Br atoms	Integer
Log(P)	Decimal
Number of H-bond acceptors	Integer
Number of H-bond donors	Integer
Number of rotatable bonds	Integer
Number of heavy atoms	Integer
Complexity (Bertz formula)	Integer
Number of defined atom stereocentres	Integer
Number of undefined atom stereocentres	Integer
Number of defined bond stereocentres	Integer
Number of undefined bond stereocentres	Integer

composed of two identically sized sets of an FC and BN layer, all followed by a dropout layer. The purpose of these dropout layers is to prevent the model from being able to ‘memorize’ the training data to artificially increase its training accuracy. The first and second latent layers as well as the last latent layer lack a dropout layer following them to prevent early-stage data loss and highly variable classification respectively. The FC layers use rectified linear units (ReLU) for activation, while the output layer uses a sigmoid function. Adam was used as the optimizer, with parameters left as default. All dropout layers use a drop rate of 0.3, and the total number of trainable parameters in the model is 744,449. Various numbers of epochs were used for training in order to identify the optimal value, and batch sizes between 32 and 256 were tested for the same purpose.

After training, the model is evaluated on the testing set to quantify how well it has learned (i.e. its generalizability). As well, the model is tested on the interactions (or lack thereof) of drugs upon which it has not been trained, which further assesses how well this model predicts drug-drug interactions. The data for this last test is obtained from Kim et al. [4].

Table 2. Layers composing the model.

Type	Activation	Output	Parameters
Input		40	
FC	ReLU	256	11520
BN		256	1024
FC	ReLU	256	65792
BN		256	1024
FC	ReLU	512	131584
BN		512	2048
FC	ReLU	512	262656
BN		512	2048
Dropout		512	
FC	ReLU	256	131328
BN		256	1024
FC	ReLU	256	65792
BN		256	1024
Dropout		256	
FC	ReLU	128	32896
BN		128	512
FC	ReLU	128	16512
BN		128	512
Dropout		128	
FC	ReLU	128	16512
BN		128	512
FC	Sigmoid	1	129

RESULTS

In this project, the architecture of the network remained fundamentally unchanged throughout testing after being determined from early tests on several variants. These early tests were focused on the number of FC layers in each super-layer, as well as the number of super-layers (i.e. the depth of the network). As well, it was determined how many FC layers were to be included before the first super-layer, and after the final. As can be seen in Table 2, it was found that the model performed optimally (in terms of the validation accuracy after 100 epochs) with two FC layers, then three super-layers, and then a single FC layer before the output. In whole, every combination of 1 to 3 FC layers before the super-layers, 1 to 5 super-layers, and 1 to 3 FC layers after the super-layers was tested. As well, different numbers of neurons in each layer and super-layer were tested, all with the diamond pattern described earlier. It was found that the sizes

Figure 1. A plot of the training and validation accuracy over 65 epochs for a model trained on approximately 608,000 samples with a batch size of 64. Training took about 120 minutes. The testing accuracy for this model is 0.965 on unseen interactions, and 0.942 on unseen drugs.

very closely with the validation accuracy at epoch 65 of 0.965. When tested on a set of unseen interactions between unknown drugs, the model scored a binary accuracy of 0.942. This small drop in accuracy is expected due to the generalization of the data being tested. Figure 2 shows a confusion matrix for predictions of drug interactions in the seen drug dataset. The model itself is relatively small, with the whole architecture and learned weights coming out to about 9 MB in size.

DISCUSSION

In training a slew of models based on the determined architecture, we were able to tune the hyperparameters (including the dropout rate, output size of FC layers, activation functions and learning rate) and ascertain the best performance from our chosen architecture. Although simplistic, the simplicity is what makes the model as efficient as it is. To train over 65 epochs on about 650,000 interactions takes just under two hours on two Nvidia® P100 GPUs; quite a short time in this field. This makes the model highly extensible to much larger datasets, while staying within reasonable training times, even without optimal hardware. The potential for models such as this to find a purpose outside dedicated research laboratories hinges on their efficiency. A model that can be accessible outside of the field with necessary computational power, ease of use, and ease of access could be used in a consumer facing application that predicts drug interactions. We believe that our model meets these criteria.

Of most important note from our results are the final accuracies this model was able to achieve for both known and unknown drugs. The latter of the two was lower, at only 0.942, but this is to be expected with the generalization of the model. The extremely small drop in accuracy (just under 0.03) greatly encourages the efficacy of this model at predicting drug-drug interactions. It seems to have been able to extract some of the most important information that determines whether two drugs will interact, as it does so without having seen those drugs in its training.

The importance of models as the one we have created here cannot be overstated. Being able to predict drug-drug interactions is vitally important in both the commercial setting of drug development, as well as in the context of an individual being able to know and understand what drugs they should and should not be able to take together. In order for a model to address the needs in whatever sector, it must be

Figure 2. A confusion matrix showing the correct and incorrect predictions of drug-drug interactions over the seen drugs testing set. The column-wise contrast of colour intensity represents the statistical power and specificity of 0.937 and 0.996 respectively.

256, 512, 256, 128, 128 performed best of the tested hyperparameters.

The final, best-performing model used the architecture mentioned previously, with uniform dropout rates of 0.3, a batch size of 64, and trained over 65 epochs. The training accuracy curve can be seen in Figure 1. Of particular note is how well the validation accuracy mimics the training accuracy. When tested on a set of unseen interactions for known drugs, the model scored a binary accuracy of 0.968, lining up

able to, with high accuracy, quickly determine from minimal information whether two compounds will interact. Where many other models working to this end have failed previously is in this last criterion [1, 4], most models require an intimate knowledge of the structures of the chemicals of interest at a minimum. In addition, they often require a host of other information that is not easily accessible for any given molecule. The strength of our minimal information requirement is that these parameters can be experimentally determined without knowing or having to encode the structure of the molecule using very readily available properties of compounds. This saves not only on the burden of procuring datasets that have all the necessary information; it also reduces the number of necessary tests to be run on a compound before one has the requisite amount of necessary information to predict the interaction of two drugs. This decreases the time to train and utilize the model thereby making it more accessible in whole.

As mentioned previously, one of the primary extensions of the model in its current state is to switch from the binary classification of interaction to a categorical classification of the type and location of interaction. We consider this a natural extension; preliminary tests with small sets indicate promising results, similar to the accuracy for binary classification. The procurement of a dataset that has labelled interactional information could be created through processing raw data provided by Kim et al. [4]. This would resolve primary barriers with respect to classification of drug interactions.

A further improvement is to increase the number of parameters read into the network for each drug. While the set given provides a relatively high degree of accuracy, the classification accuracy could be improved by including other, easily obtainable properties of the chemicals of interest. One clear boundary for this is the time it would take to compile a labelled dataset with the requisite information. The use of minimally-many inputs may also contribute to a higher rate of false positives and negatives due to increased ambiguity between compounds over a more detailed model.

Finally, we would like to perform a sensitivity analysis on the model. This is an endeavour in and of itself, but doing so would allow for an entirely new and informative purpose to the model by correlating each input with its effect on the model's output. This is effectively a way to determine which chemical properties are most important in determining whether or not they will interact in the body. The hope in doing so is to further investigate where areas of research should focus to

best characterize and predict these interactions.

ACKNOWLEDGEMENTS

This research was carried out with support and guidance from STEM Fellowship. The authors acknowledge data resources from the SEES Lab, DrugBank, and PubChem. The authors were recognized for their research efforts by monetary awards funded by Roche and Schulich as part of the *Undergraduate Big Data Challenge*.

CONFLICT OF INTEREST

The authors declare no conflicts of interest.

REFERENCES

1. Kastrin A, Ferk P, Leskošek B. Predicting potential drug-drug interactions on topological and semantic similarity features using statistical learning. PLoS ONE. 2018 May 8; 13(5): e0196865. Available from: <https://doi.org/10.1371/journal.pone.0196865>
2. Cami A, Manzi S, Arnold A, Reis BY. Pharmacointeraction network models predict unknown drug-drug interactions. PLoS ONE. 2013 Apr 19; 8(4):e61468. DOI: 10.1371/journal.pone.0061468.
3. Science and Engineering of Emerging Systems Lab. (2012). 2012 snapshot of the DrugBank database. Available from: <http://seeslab.info/downloads/drug-drug-interactions/>
4. Kim S, Chen J, Cheng T, Gindulyte A, He J, He S, et al. PubChem 2019 update: improved access to chemical data. Nucleic Acids Res. 2019 Jan 8; 47(D1):D1102-1109. DOI: 10.1093/nar/gky1033. [PubMed PMID: 30371825]
5. Takeda T, Hao M, Cheng T, Bryant SH, Wang Y. Predicting drug-drug interactions through drug structural similarities and interaction networks incorporating pharmacokinetics and pharmacodynamics knowledge. J Cheminform. 2017 Mar 7; 9:16. DOI: 10.1186/s13321-017-0200-8. PubMed PMID: 28316654; PubMed Central PMCID: PMC5340788.