Research

Classification of drugs based on mechanism of action using machine learning techniques

H. L. Gururaj[1] · Francesco Flammini[2] · H. A. Chaya Kumari[1] · G. R. Puneeth[1] · B. R. Sunil Kumar[1]

Received: 15 September 2021 / Accepted: 18 October 2021
Published online: 24 November 2021

Abstract
The mechanism of action is an important aspect of drug development. It can help scientists in the process of drug discovery. This paper provides a machine learning model to predict the mechanism of action of a drug. The machine learning models used in this paper are Binary Relevance K Nearest Neighbors (Type A and Type B), Multi-label K-Nearest Neighbors and a custom neural network. These machine learning models are evaluated using the mean column-wise log loss. The custom neural network model had the best accuracy with a log loss of 0.01706. This neural network model is integrated into a web application using Flask framework. A user can upload a custom testing features dataset, which contains the gene expression and the cell viability levels. The web application will output the top classes of drugs, along with the scatter plots for each of the drug.

Keywords Mechanism of action · Cell viability · Gene expression · Protein · Inhibitors

1 Introduction

The term "mechanism of action" (MoA) refers to how a medicine or other substance causes an effect in the body. A drug's mechanism of action, for example, could be how it affects a specific target in a cell, such as an enzyme, or how it impacts a cell function, such as cell proliferation. Knowing a drug's mechanism of action can provide information about the drug's safety and how it affects the body [1].

The majority of medications work by interacting with proteins in the host or pathogen. Drug targets include a variety of proteins, and the name receptor is only used when the interaction results in a signal transmission cascade. A receptor is a molecule or polymeric structure that identifies and binds an endogenous substance on the surface or inside a cell. When a substance elicits a detectable physiological or pharmacological response characteristic of the receptor, it is said to be an agonist. Some medications may be unable to initiate any action on their own after binding to a receptor site, but they can prevent the action of other agonists. These are called Antagonists [2].

Understanding a biologically active compound's mechanism of action entails not only identifying the target but also investigating the biological chemistry that occurs before or after target binding. Many genes are involved in a drug's mechanism of action, and so have an impact on sensitivity. The mechanism of action of a small molecule encompasses both the intracellular target(s) and the activities that occur before and after target engagement [3].

Because of the high level of intricacy in the interactions between the Tuberculosis medicine and Mycobacterium tuberculosis, Tuberculosis treatment requires an adequate cognizance of MOA, which is critical for the successful delivery

✉ H. L. Gururaj, gururaj1711@vvce.ac.in | [1]Department of CSE, Vidyavardhaka College of Engineering, Mysore, Karnataka, India. [2]Sciences and Arts of Southern Switzerland (CH), University of Applied, Manno, Switzerland.

Discover Artificial Intelligence (2021) 1:13 | https://doi.org/10.1007/s44163-021-00012-2

of drug candidates. Several methods were outlined for investigating tuberculosis drug MoA's and provided guidance for future tuberculosis drug development. In the context of Mycobacterium tuberculosis pathogenesis, they evaluated diverse platforms for their strengths and limitations in Tuberculosis drug MOA elucidation [4].

Bioinformatics is a science that uses several layers of information such as image-based data, pathways and gene expression to aid in the understanding of Mechanisms of Action. In order to comprehend MoA, it is necessary to analyse the complicated responses of the human biological system to drug treatments. The impact of bio-informatics on drug discovery and several Bioinformatic methods for the comprehension of Mechanisms of Action were discussed [5].

This paper discusses about the various machine learning models and their accuracies. Further, a web application is developed using the Flask web framework. The machine learning model which has the best accuracy is utilized for developing this web application. This web application can be helpful for the scientists in the drug discovery process.

The rest of this paper is organized as follows. In Sect. 2, a literature survey of mechanism of action is described. In Sect. 3, the methodology is shown. In Sect. 4, the pre-processing of the dataset is presented. In Sect. 5, the evaluation of the machine learning model is elucidated. In Sect. 6, various machine learning models and their results are presented. In Sect. 7, the architecture of the web application is shown. In Sect. 8, the snapshots of the running web application is presented. Finally, Sect. 9 concludes this paper.

2 Literature survey of mechanism of action

Some drugs's mechanism of action are unrevealed. Meanwhile, several drugs' mechanism of action have been already discovered. For example, Aspirin's mechanism of action involves irreversible inhibition of the enzyme cyclooxygenase, which reduces inflammation and pain by decreasing the formation of thromboxanes and prostaglandins.

Different drugs can have different mechanism of action. A literature survey of mechanism of action of various drugs are described in Table 1.

Table 1 Literature survey of mechanism of action

Title	Findings
Efficacy and Mechanism of Action of Marine Alkaloid 3,10-Dibromofascaplysin in Drug-Resistant Prostate Cancer Cells [6]	In human prostate cancer cells with various levels of treatment resistance, the efficacy and mechanism of action of the marine alkaloid 3,10-dibromofascaplysin were examined. All of the cell lines tested showed anticancer activity
The mechanism of action of aspirin [7]	The enzyme cyclooxygenase is inhibited by aspirin. Aspirin-like medications hindered the generation of physiologically significant Prostaglandins by blocking this step in the Prostaglandin production process. This gave a unified explanation for the aspirin-like medicines' therapeutic activity and its common adverse effects
Research on the Mechanism of Action of a Citrinin and Anti-Citrinin Antibody Based on Mimotope X27 [8]	A mimotope is a powerful recognition receptor that can be utilised to investigate antigen and antibody mechanisms of action. The mimotope method was used to develop a binding model between citrinin and antibody. They discussed a method for increasing the sensitivity of detection of citrinin in immunoassays
The mechanism of action of ramoplanin and enduracidin [9]	They tested whether ramoplanin and enduracidin had an innate preference for one step over the other using inhibitory kinetics and binding assays. They discovered that as compared to the MurG stage, both ramoplanin and enduracidin hindered the transglycosylation step of peptidoglycan production
Mechanism of action of antipruritic drugs[10, 18]	They found that Antipruritic drugs acted centrally via a sedative characteristic, but H1 receptor antagonists only have a peripheral antipruritic effect when itching is caused by histamine release
Mechanism of Action of Atypical Antipsychotic Drugs in Mood Disorders [11, 19, 20]	The neural mechanisms of existing atypical antipsychotics and prospective antipsychotics were discussed in this paper, as well as how they relate to their efficacy in mood disorders such as anxiety and depression

3 Methodology

Serialization and de-serialization is used in order to avoid training the model using the training dataset every time a fresh testing dataset is submitted to the web application.

As shown in the Fig. 1, in the first step, the model is trained using the training dataset. This model is saved into a file, which is present in the HDF5 format [12]. HDF5 stands for Hierarchical Data Format 5. This process is called serialization and this is done using the keras built-in module "save". The model's architecture, weights, training setup (loss and optimizer), and optimizer state are all contained in this serialized file.

Later, this file is de-serialized and it is supplied into the Flask web application. This process is called de-serialization and this is done using the keras built-in module "load_model". In this way, whenever a new testing dataset is uploaded in the web application, training the model with the training dataset every single time can be avoided (the model uses the serialized file to load the pre-trained model's configuration).

4 Data pre-processing

A detailed analysis of this dataset and the methodology used has been provided in a previous paper [13]. The dataset is taken from kaggle [14]. As shown in Fig. 2, first, the given dataset is divided into the training dataset and the testing dataset. Furthermore, the training dataset and the testing dataset have been divided into the features dataset and the target dataset.

Both the training features dataset and the training target dataset consists of 23,814 training samples. Also, both the testing features dataset and the testing target dataset consists of 3982 testing samples. In the data pre-processing stage, the categorical values of the attributes are mapped into numerical values as shown in Table 2.

Fig. 1 Methodology of the machine learning model

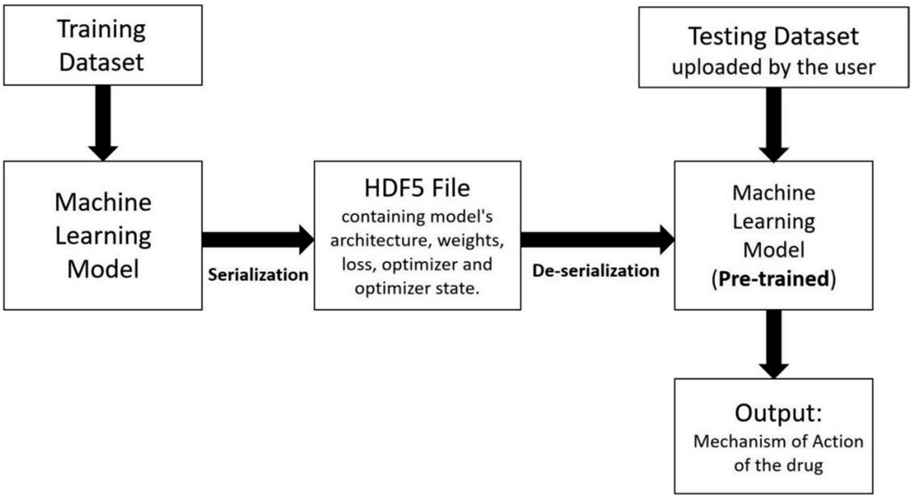
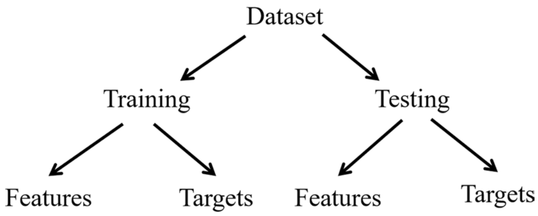

Training Dataset
Testing Dataset uploaded by the user
Machine Learning Model
Serialization
HDF5 File containing model's architecture, weights, loss, optimizer and optimizer state.
De-serialization
Machine Learning Model (Pre-trained)
Output: Mechanism of Action of the drug

Fig. 2 Splitting of the dataset



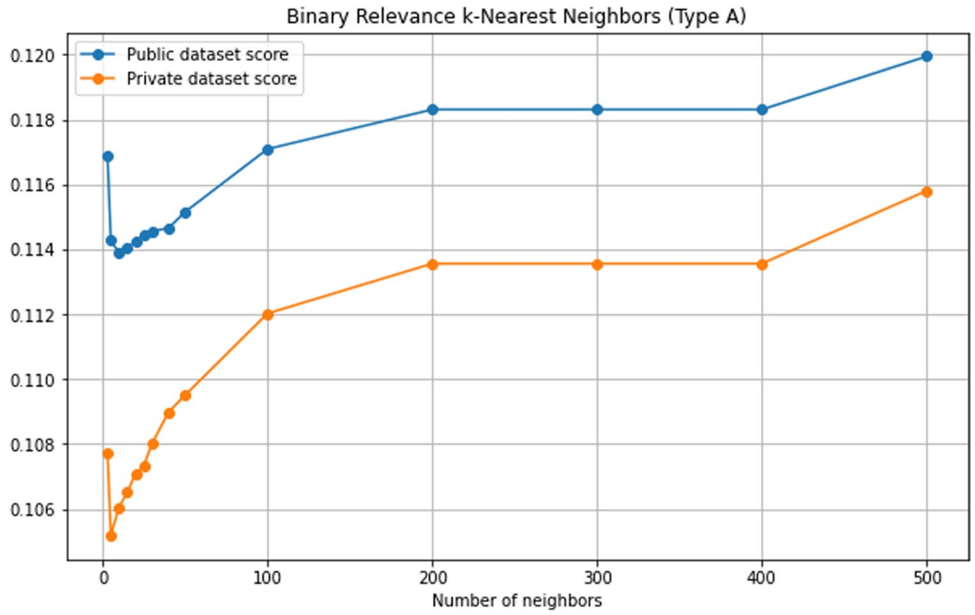


Fig. 3 Graph representing the accuracy of the BRkNN-a model

5 Evaluation of the machine learning model

The accuracy of the machine learning model is evaluated by applying log loss function for each drug-MoA annotation pair. The mean column-wise log loss is used for the evaluation of the model.

For every sample id "sig_id", the probability that the sample had a positive response needs to be predicted for each MoA target. A positive response means that a drug belongs to a particular class of drug (i.e. target). A lesser value of log loss (i.e. score) indicates better accuracy.

The formula for the evaluation of the machine learning model is shown in Eq. (1) [15].

$$\text{score} = -\frac{1}{M}\sum_{m=1}^{M}\frac{1}{N}\sum_{i=1}^{N}\left[y_{i,m}\log\left(y_{i,m}\right) + \left(1 - y_{i,m}\right)\log\left(1 - y_{i,m}\right)\right], \tag{1}$$

where: N is the number of sig_id observations in the test data (i = 1,2,…,N). M is the number of scored MoA targets (m = 1,2,…,M). $\hat{y}_{i,m}$ is the predicted probability of a positive MoA response for a sample id (sig_id). $y_{i,m}$ is the ground truth, 1 for a positive response, 0 otherwise. log() is the natural base e logarithm.

6 Models and results

In the dataset, there could be numerous Mechanism of Action's (MoA's) for each of the drug. Thus, this machine learning problem belongs to multi-label classification. The machine models tested in this paper are BRkNN (Binary Relevance K Nearest Neighbors), ML-KNN (Multi-label K-Nearest Neighbors) and a custom Neural Network.

Table 2 Mapping of the attribute's values

Attribute name	Old value	New value
cp_type	trt_cp	0
	ctl_vehicle	1
cp_time	24 h	0
	48 h	1
	72 h	2
cp_dose	D1	0
	D2	1

6.1 BRKNN (binary relevance K nearest neighbors)

BRkNN is a variant of the k-Nearest Neighbors (kNN) method that is essentially equal to combining Binary Relevance (BR) with the kNN algorithm. BRkNN extends the kNN method to make independent predictions for each label [16].

Based on the assessment of each label's confidence score, BRKNN is classified into two types: BRkNN-a and BRkNN-b.

6.1.1 BRkNN-a (type A)

BRkNN-a determines whether BRkNN returns the empty set if none of the labels appear in at least half of the k nearest neighbours. If this criterion is met, the label with the highest confidence is outputted [16]. For this model, Fig. 3 and Table 3 shows the graph and prediction scores, respectively.

In the graph shown in Fig. 3, the X-axis represents the number of neighbours, and the Y-axis represents the public and private dataset score. The private dataset score improves from 3 neighbours until 5 neighbours. The public dataset score improves until from 3 neighbours until 10 neighbours. Afterwards, both of the scores decline as the number of neighbours increases.

6.1.2 BRkNN-b (type B)

First, BRkNN-b estimates the "s" (average size) of the label sets of the k nearest neighbors, and later, outputs the integer which is nearest to "s" labels, which is having the highest confidence [16]. For this model, Fig. 4 and Table 4 shows the graph and prediction scores, respectively. In the graph shown in Fig. 4, the X-axis represents the number of neighbors and the Y-axis represents the public and private dataset score. Both the private dataset score and the public dataset score improves from 3 neighbors until 2000 neighbors. 5000 neighbors yeilds a worse score. Afterwards, both of the scores remains constant. As seen from Table 4, the maximum difference between the public dataset score and private dataset score is 0.38482, which occurs when the number of neighbors is 30.

6.2 ML-KNN (multi-label K-nearest neighbors)

The ML-KNN technique is based on the well-known k-Nearest Neighbor (kNN) algorithm. First, the k nearest neighbors in the training set are selected for each test instance. The maximum a posteriori (MAP) concept is then used

Table 3 Prediction scores of the BRkNN-a model

Number of neighbours	Private dataset score	Public dataset score
3	0.10773	0.11688
5	0.10518	0.11427
10	0.10605	0.11389
15	0.10651	0.11406
20	0.10709	0.11423
25	0.10733	0.11444
30	0.10801	0.11456
40	0.10898	0.11465
50	0.1095	0.11515
100	0.11202	0.11709
200	0.11356	0.11831
300	0.11356	0.11831
400	0.11356	0.11831
500	0.11581	0.11995

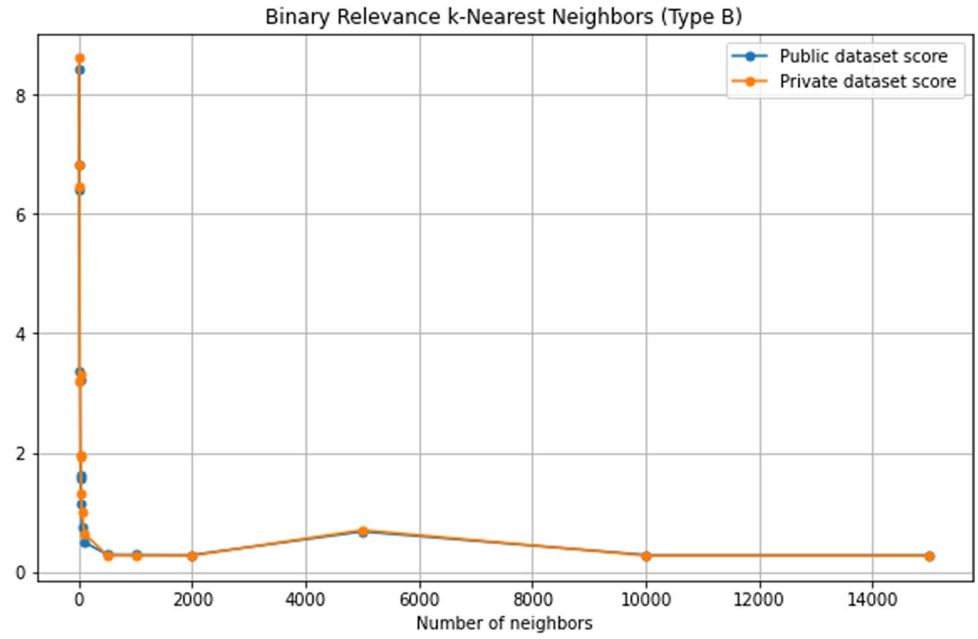


Fig. 4 Graph representing the accuracy of the BRkNN-b model

Table 4 Prediction scores of the BRkNN-b model

Number of neighbors	Private dataset score	Public dataset score	Difference between private and public score
3	8.61531	8.44115	0.17416
5	6.82411	6.83867	− 0.01456
10	6.46235	6.41718	0.04517
15	3.21137	3.36569	− 0.15432
20	3.30373	3.2381	0.06563
25	1.92898	1.63633	0.29265
30	1.96073	1.57591	0.38482
40	1.30347	1.15522	0.14825
50	0.99511	0.76119	0.23392
100	0.6386	0.49542	0.14318
500	0.27721	0.28901	− 0.0118
1000	0.27465	0.28606	− 0.01141
2000	0.27254	0.28282	− 0.01028
5000	0.70056	0.67626	0.0243
10,000	0.27254	0.28282	− 0.01028
15,000	0.27254	0.28282	− 0.01028

to determine the label set for the test instance based on statistical information obtained from the label sets of the neighbouring examples [17]. For this model, Fig. 5 and Table 5 shows the graph and prediction scores, respectively. In the graph shown in Fig. 5, the X-axis represents the number of neighbors and the Y-axis represents the public and private dataset score. Both the private dataset score and the public dataset score improves from 3 neighbors until 20 neighbors. Afterwards, both of the scores declines as the number of neighbors increases.

6.3 Custom neural network

A neural network is created using Keras [18–20]. Keras is a Python-based deep learning API that runs on top of TensorFlow. Since there are 875 input features in the dataset, the input layer units is 875. Similarly, there are 206 output

Fig. 5 Graph representing the accuracy of the ML-KNN model

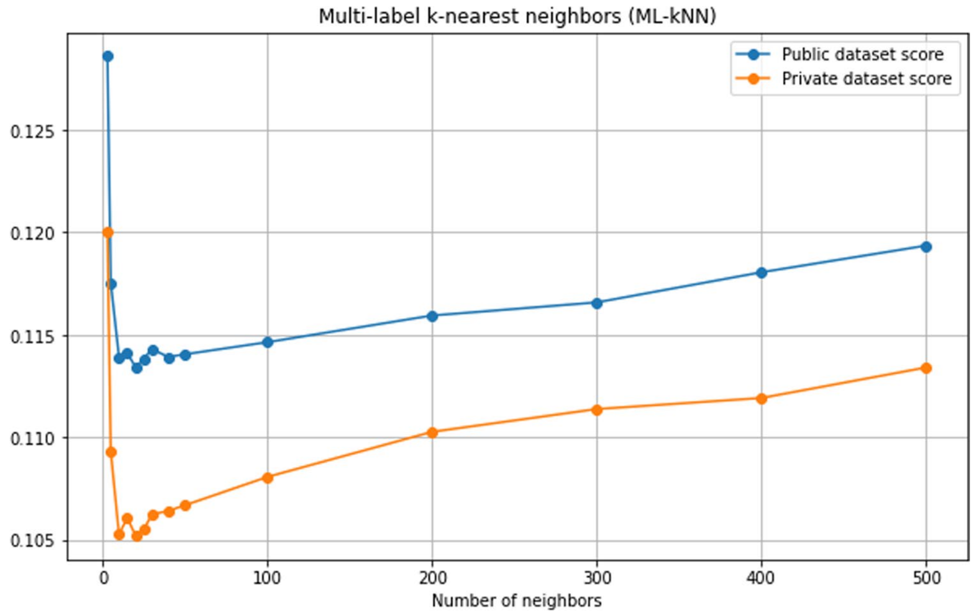


Table 5 Prediction scores of the ML-KNN model

Number of neighbors	Private dataset score	Public dataset score
3	0.12005	0.12859
5	0.10933	0.11755
10	0.10528	0.11393
15	0.10609	0.11414
20	0.10523	0.11343
25	0.10552	0.11385
30	0.10625	0.11431
40	0.10643	0.11393
50	0.10670	0.11406
100	0.10808	0.11465
200	0.11028	0.11595
300	0.11139	0.11659
400	0.11193	0.11806
500	0.11342	0.11936

targets, the output layer units is 206. Both the dropout layer 1 and dropout layer 2 have 0.5 as the dropout rate. The model is compiled using the binary cross-entropy loss function. The optimizer used is adam. The Neural network implementation code can be found in Github [21]. Figure 6 shows the layers of the neural network. Table 6 gives the description of the each of the layer used. Table 7 shows the activation functions used for the dense layers and the output layer. Figures 7 and 8 shows the graph for the sigmoid and RELU activation function, respectively. Figure 9 shows the accuracy graph. Table 8 shows the prediction scores for this model.

In the graph shown in Fig. 9, the X-axis represents the epochs and the Y-axis represents the public and private dataset score. Both the private dataset score and the public dataset score improves from 15 epochs until 75 epochs. Afterwards, both of the scores declines as the number of epochs increases.

Fig. 6 Layers of the custom neural network

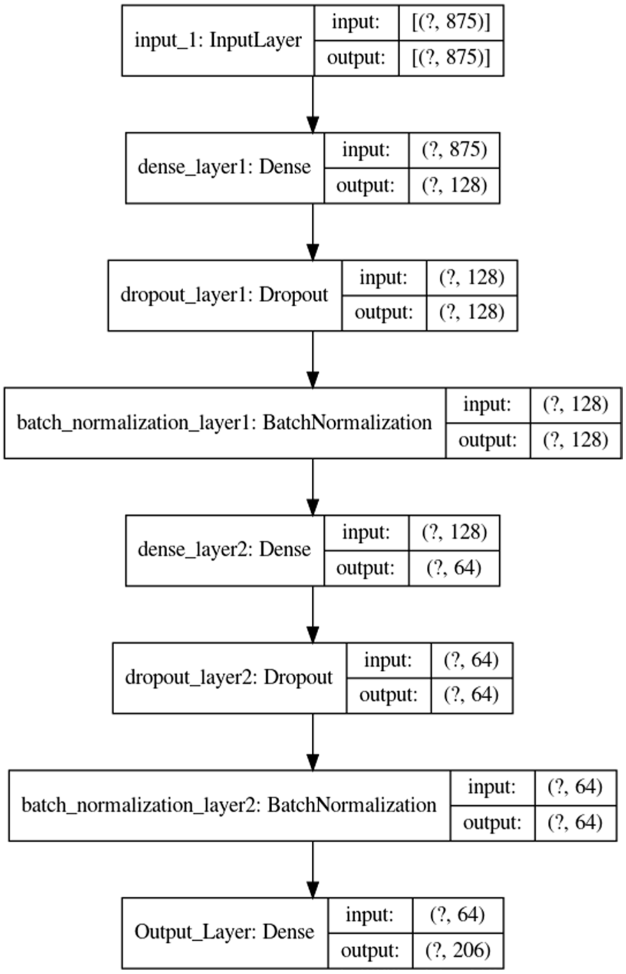


Table 6 Description of the layers of the custom neural network

Layer	Output shape	Number of parameters
Input layer	(None, 875)	0
Dense layer 1	(None, 128)	112,128
Dropout layer 1	(None, 128)	0
Batch normalization layer 1	(None, 128)	512
Dense layer 2	(None, 64)	8256
Dropout layer 2	(None, 64)	0
Batch normalization layer 2	(None, 64)	256
Output layer	(None, 206)	13,390

Table 7 Activation function used for the custom neural network

Layer	Activation function
Dense layer 1	RELU
Dense layer 2	RELU
Output layer	Sigmoid

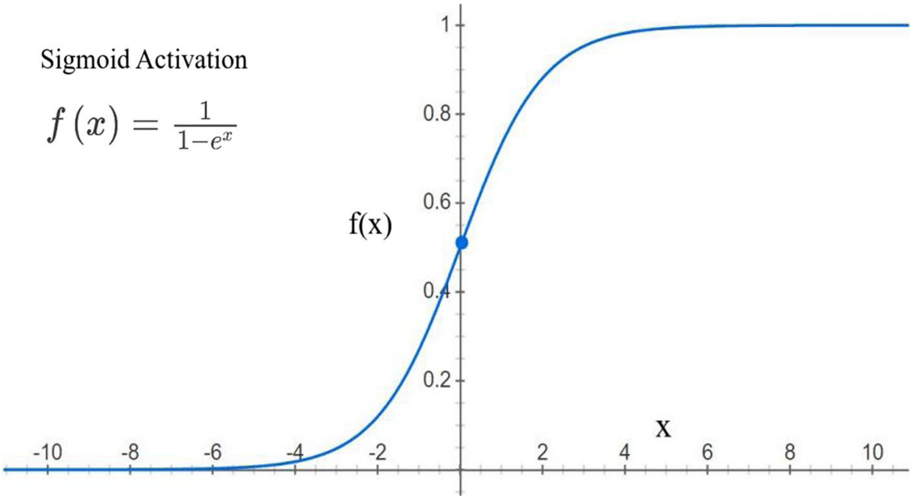


Fig. 7 Graph for sigmoid activation function

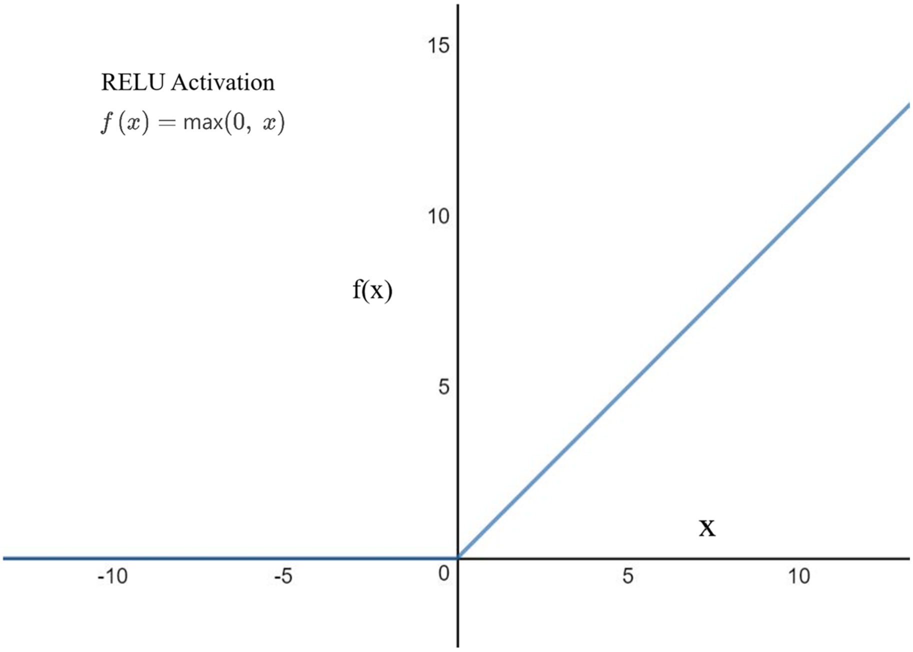


Fig. 8 Graph for RELU activation function

Since the Private Dataset Score is considered for scoring the final leader board, the best score for private dataset obtained with each of the models is considered. The summary of the best accuracy for each of the model is shown in the Table 9. As seen from the Table 9, the custom neural network with 75 epochs and 100 batch size performs the best.

7 Architecture of the web application

A web application is developed to visualize the Mechanism of Action of each of the drug. The source code for this web application can be found in Github [21]. This web application is developed using the Flask framework [22]. It also uses the Jinga Templating Engine [23].

Figure 10 shows the architecture of the web application. The input is a CSV file, which contains gene expression and cell viability values of each drug. The output is the top classes of drugs (with the highest probability). The number of top MoA's to be displayed in the output can be set using the variable NUMBER_OF_TOP_MOA.

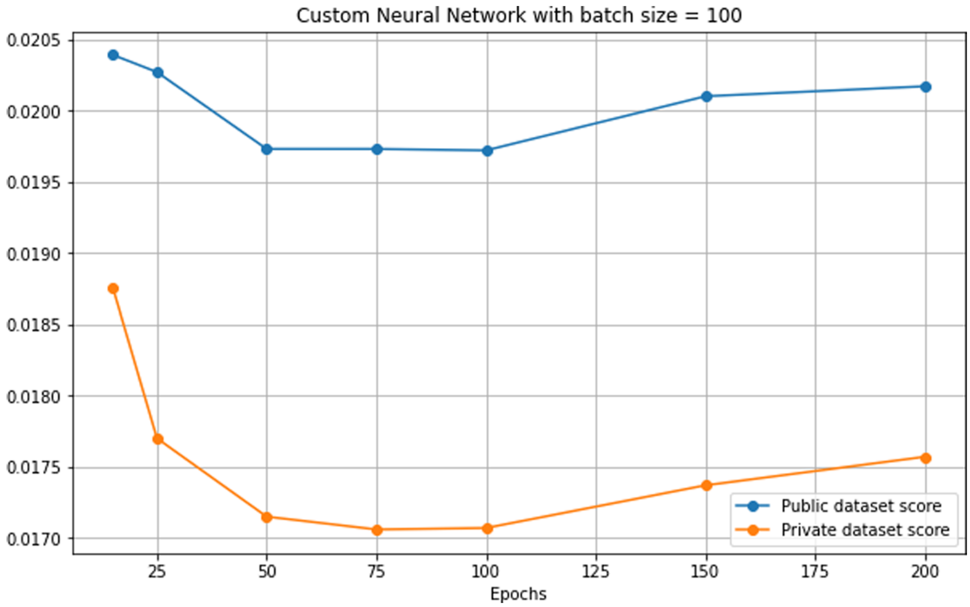


Fig. 9 Graph representing the accuracy of the custom neural network

Table 8 Prediction scores of the custom neural network

Epochs	Private dataset score	Public dataset score
15	0.01875	0.02039
25	0.0177	0.02027
50	0.01715	0.01973
75	0.01706	0.01973
100	0.01707	0.01972
150	0.01737	0.0201
200	0.01757	0.02017

Table 9 Summary of the best prediction scores for each of the model

Model name	Best private dataset score	Configuration
BRkNN-a	0.10518	5 Neighbors
BRkNN-b	0.27254	2000 Neighbors
ML-KNN	0.10523	20 Neighbors
Custom neural network	0.01706	75 Epochs, 100 Batch size

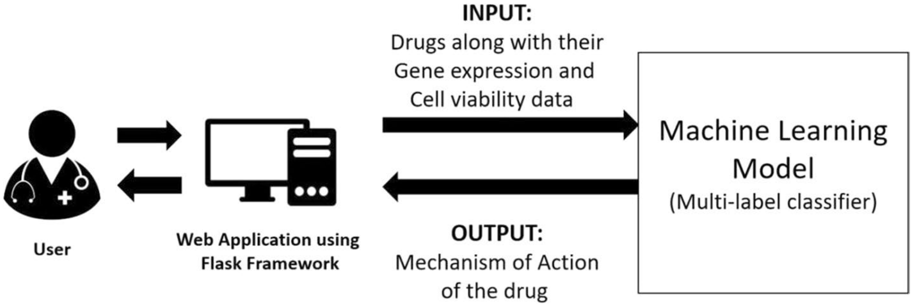

INPUT:
Drugs along with their Gene expression and Cell viability data
Machine Learning Model
(Multi-label classifier)
User
Web Application using Flask Framework
OUTPUT:
Mechanism of Action of the drug

Fig. 10 Web application's architecture

8 Running the web application

In Table 10, the testing dataset which is to be uploaded in the web application (testing_dataset.csv) is shown. For each of the drug present in the testing dataset, the top MoA's for each of the drug is displayed. Along with it, a scatter plot is displayed for each of the drug. In order to visualize the scatter plot, an ID is given to each of the class of drug. The first class of drug (i.e. 5-alpha_reductase_inhibitor) is given ID 1, the second class of drug (i.e. 11-beta-hsd1_inhibitor) is given ID 2 and so on. This list of ID's is stored in a python list (present in the flask application). The X-axis of the scatter plot represents the ID of the class of the drug, and the Y-axis represents the probability of the drug belonging to that class of the drug. The homepage of the web application is shown in Fig. 11. The top classes of drug is shown in Figs. 12 and 13. The scatter plot is shown in Figs. 14 and 15.

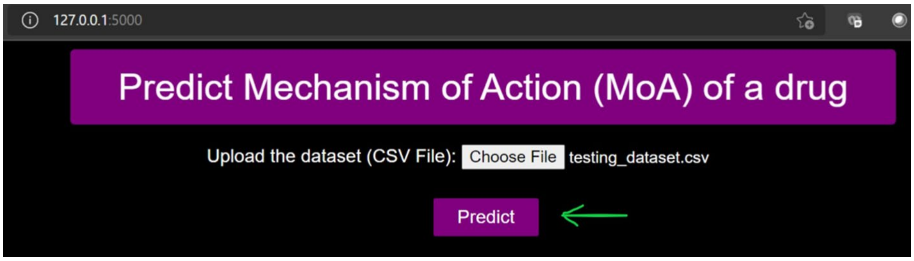
9 Conclusion

Mechanism of drug can help scientists accelerate the drug discovery process. This paper discussed various machine learning models to predict the mechanism of action of a drug. Also, a Flask-based web application is introduced, through which a user can input a custom testing features data-set containing gene expression and cell viability levels. The output produced is the top classes of drugs, along with their scatter plot. This can help scientists to predict the Mechanism of Action and can help them in the discovery of new drugs.

Figure 16 and Table 11 shows a summary regarding the log loss of all the machine learning models used in this paper. Among these models, the custom neural network model performed the best.

Table 10 Example testing dataset to be uploaded in the web application (contains two drugs)

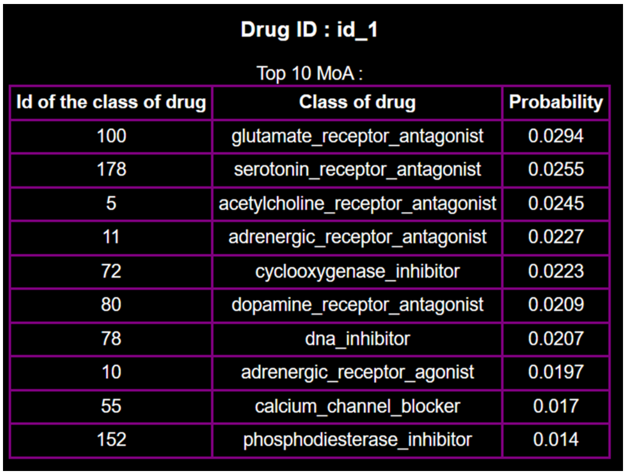
sig_id	cp_type	cp_time	cp_dose	g-0	…	g-771	c-0	…	c-99
id_1	trt_cp	48	D2	0.3707	…	−1.4167	– 1.3178	…	– 1.4598
id_2	trt_cp	72	D1	– 0.4631	…	0.1453	– 0.4151	…	0.5966
id_3	ctl_vehicle	24	D2	0.4876	…	– 1.7308	– 2.2818	…	– 0.5757



Predict Mechanism of Action (MoA) of a drug

Upload the dataset (CSV File): testing_dataset.csv

Fig. 11 Home page of the web application



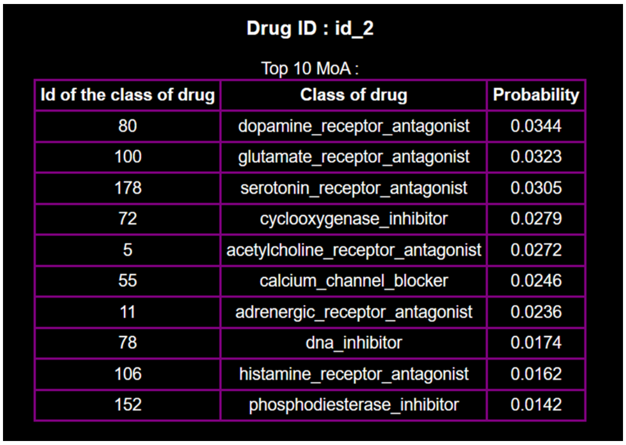
Drug ID : id_1

Top 10 MoA :

Id of the class of drug	Class of drug	Probability
100	glutamate_receptor_antagonist	0.0294
178	serotonin_receptor_antagonist	0.0255
5	acetylcholine_receptor_antagonist	0.0245
11	adrenergic_receptor_antagonist	0.0227
72	cyclooxygenase_inhibitor	0.0223
80	dopamine_receptor_antagonist	0.0209
78	dna_inhibitor	0.0207
10	adrenergic_receptor_agonist	0.0197
55	calcium_channel_blocker	0.017
152	phosphodiesterase_inhibitor	0.014

Fig. 12 Top classes of drug along with their probability (for the first drug)

Fig. 13 Top classes of drug along with their probability (for the second drug)

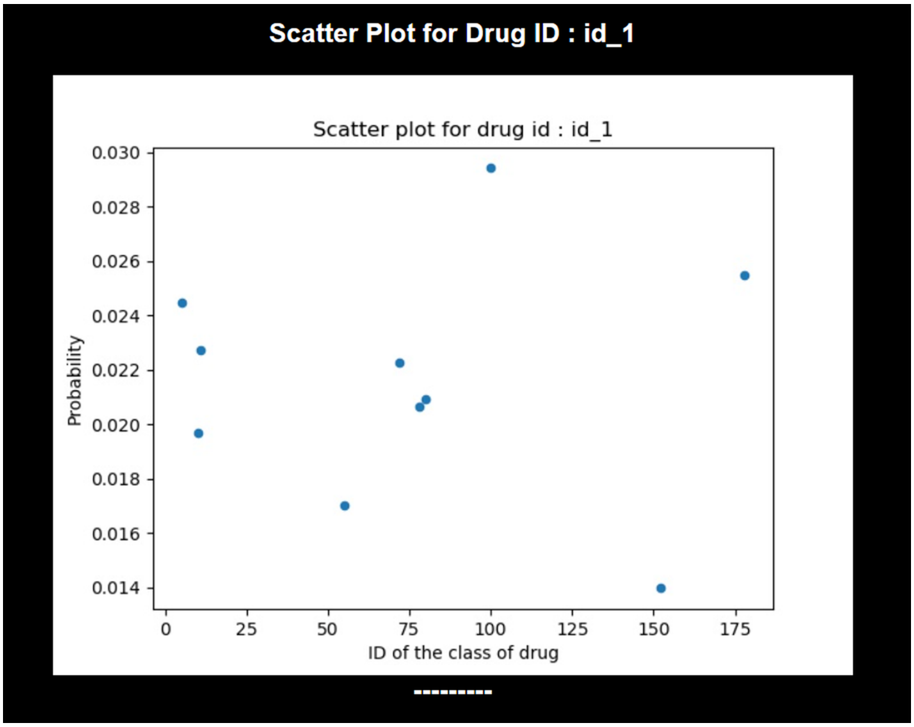


Drug ID : id_2

Top 10 MoA :

Id of the class of drug	Class of drug	Probability
80	dopamine_receptor_antagonist	0.0344
100	glutamate_receptor_antagonist	0.0323
178	serotonin_receptor_antagonist	0.0305
72	cyclooxygenase_inhibitor	0.0279
5	acetylcholine_receptor_antagonist	0.0272
55	calcium_channel_blocker	0.0246
11	adrenergic_receptor_antagonist	0.0236
78	dna_inhibitor	0.0174
106	histamine_receptor_antagonist	0.0162
152	phosphodiesterase_inhibitor	0.0142

Fig. 14 Scatter plot (for the first drug)



Scatter Plot for Drug ID : id_1

In the future, if a better-performing machine learning model is discovered, they can seamlessly integrate their machine learning model into the web application provided in this paper. Also, for a better visualization of the scatterplots, the web application can be extended to use interactive scatterplots using plotly, bokeh or any other plotting packages.

Fig. 15 Scatter plot (for the second drug)

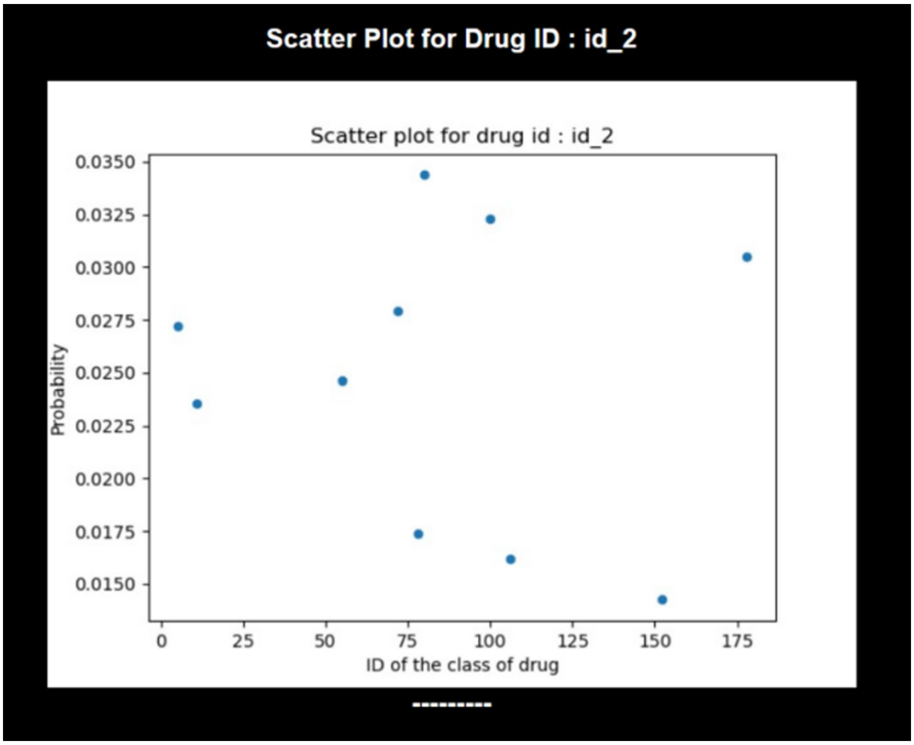


Fig. 16 Bar chart representing the log loss of machine learning models

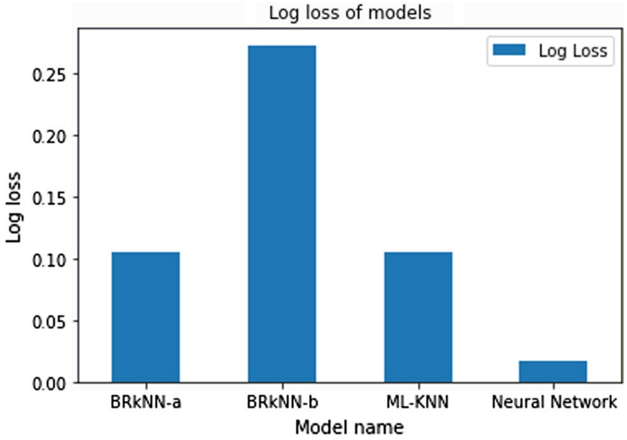


Table 11 Log loss of all machine learning models

Model name	Log loss
BRkNN-a	0.10518
BRkNN-b	0.27254
ML-KNN	0.10523
Custom Neural Network	0.01706

Author contributions HLG, CHA wrote the main manuscript text. PGR, SBR designed the model and applied suitable ML algorithmsto draw the results. FF validated the data sets and results. HLG, CHA proof read the article before submitting. All authors read and approved the final manuscript.

Declarations

Competing interests The authors declare no competing interests.

References

1. Definition of mechanism of action, National Cancer Institute, https://www.cancer.gov/publications/dictionaries/cancer-terms/def/mechanism-of-action.
2. Pierre-Louis T. Mechanism of drug action and pharmacokinetics/pharmacodynamics integration in dosage regimen optimization for veterinary medicine. Veterinary Pharmacology and Therapeutics. Wiley. p 1525, 2018, 9781118855829. hal-02787306.
3. Palmer A. The many genes of drug mechanism. Nat Chem Biol. 2016;12:57–8. https://doi.org/10.1038/nchembio.2010.
4. Yuan T, et al. The pursuit of mechanism of action: uncovering drug complexity in TB drug discovery. RSC chemical biology. 2021;2(2):423–40. https://doi.org/10.1039/d0cb00226g.
5. Trapotsi M-A, Barrett I, Engkvist O, Bender A. Bioinformatic approaches in the understanding of mechanism of action (MoA). In: Plowright AT, editor. Target discovery and validation. https://doi.org/10.1002/9783527818242.ch11.
6. Dyshlovoy SA, et al. Efficacy and mechanism of action of marine alkaloid 3,10-dibromofascaplysin in drug-resistant prostate cancer cells. Mar Drugs. 2020;18(12):609. https://doi.org/10.3390/md18120609.
7. Vane JR, Botting RM. The mechanism of action of aspirin. Thromb Res. 2003;110(5–6):255–8. https://doi.org/10.1016/s0049-3848(03)00379-7.
8. Li Y, et al. Research on the mechanism of action of a citrinin and anti-citrinin antibody based on mimotope X27. Toxins. 2020;12(10):655. https://doi.org/10.3390/toxins12100655.
9. Fang X, et al. The mechanism of action of ramoplanin and enduracidin. Mol bioSystems. 2006;2(1):69–76. https://doi.org/10.1039/b515328j.
10. Krause L, Shuster S. Mechanism of action of antipruritic drugs. Br Med J Clin Res Ed. 1983;287(6400):1199–200. https://doi.org/10.1136/bmj.287.6400.1199.
11. Grinchii D, Eliyahu D. Mechanism of action of atypical antipsychotic drugs in mood disorders. Int J Mol Sci. 2020;21(24):9532. https://doi.org/10.3390/ijms21249532.
12. Koranne, S. Hierarchical data format 5: HDF5. In: Handbook of open source tools. Springer, Boston, MA, 2011. p. 191–200. https://doi.org/10.1007/978-1-4419-7719-9_10.
13. Puneeth GR, et al. Analysis of drug classification using mechanism of action. J Phys Conf Ser. 2021;1914(1):01204. https://doi.org/10.1088/1742-6596/1914/1/012034.
14. Mechanism of Action Dataset, Kaggle, https://www.kaggle.com/c/lish-moa/data.
15. Evaluation of the model, Kaggle, https://www.kaggle.com/c/lish-moa/overview/evaluation.
16. Spyromitros E, Tsoumakas G, Vlahavas I. An empirical study of lazy multilabel classification algorithms. In: Darzentas J, Vouros GA, Vosinakis S, Arnellos A. editors, Artificial intelligence: theories, models and applications. SETN 2008. Lecture Notes in Computer Science, vol. 5138. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-87881-0_40.
17. Min-Ling Z, Zhi-Hua Z. ML-KNN: a lazy learning approach to multi-label learning. Pattern Recogn. 2007;40(7):2038–48. https://doi.org/10.1016/j.patcog.2006.12.019.
18. Szymański P, Kajdanowicz T. A Scikit-based python environment for performing multi-label classification. 2017, allrXiv:1702.01460. arXiv.org e-Print archive. hps://arxiv.org/abs/1702.01460.
19. Ketkar N. Introduction to Keras. In: Deep learning with python. 2017. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-2766-4_7.
20. Chollet F et al. Keras. GitHub. Retrieved from https://github.com/fchollet/keras. 2015.
21. Source code for the MoA web application and the implementation of Neural network, GitHub. Retrieved from https://github.com/Puneethgr/mechanism-of-action.
22. Flask Web Framework, Official Documentation, https://flask.palletsprojects.com/.
23. Jinga Template, Official Documentation, https://jinja.palletsprojects.com/.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.