PLOS COMPUTATIONAL BIOLOGY

RESEARCH ARTICLE

Leveraging high-throughput screening data, deep neural networks, and conditional generative adversarial networks to advance predictive toxicology

Adrian J. Green[1], **Martin J. Mohlenkamp**[2], **Jhuma Das**[3], **Meenal Chaudhari**[4], **Lisa Truong**[5], **Robyn L. Tanguay**[5], **David M. Reif**[1]*

1 Department of Biological Sciences, and the Bioinformatics Research Center, NC State University, Raleigh, North Carolina, United States of America, **2** Department of Mathematics, Ohio University, Athens, Ohio, United States of America, **3** Marsico Lung Institute, University of North Carolina at Chapel Hill, Chapel Hill, North Carolina, United States of America, **4** Department of Computational Science and Engineering, North Carolina A&T State University, Greensboro, North Carolina, United States of America, **5** Department of Environmental and Molecular Toxicology, Oregon State University, Corvallis, Oregon, United States of America

* dmreif@ncsu.edu

Citation: Green AJ, Mohlenkamp MJ, Das J, Chaudhari M, Truong L, Tanguay RL, et al. (2021) Leveraging high-throughput screening data, deep neural networks, and conditional generative adversarial networks to advance predictive toxicology. PLoS Comput Biol 17(7): e1009135. https://doi.org/10.1371/journal.pcbi.1009135

Editor: Vassily Hatzimanikatis, Ecole Polytechnique Fédérale de Lausanne, SWITZERLAND

Received: October 27, 2020

Accepted: May 31, 2021

Published: July 2, 2021

Data Availability Statement: All relevant data are linked/cited within the manuscript and its Supporting Information files. We have created a new GitHub page with all code and documented examples at https://github.com/ajgreen4/Go-GAN-ZT.

Funding: This research was supported by the National Institutes of Health (NIH) grant awards ES030287 (RLT, LT), ES030007 (AJG, DMR), ES025128 (DMR), and CA161608 (AJG, DMR).

Abstract

There are currently 85,000 chemicals registered with the Environmental Protection Agency (EPA) under the Toxic Substances Control Act, but only a small fraction have measured toxicological data. To address this gap, high-throughput screening (HTS) and computational methods are vital. As part of one such HTS effort, embryonic zebrafish were used to examine a suite of morphological and mortality endpoints at six concentrations from over 1,000 unique chemicals found in the ToxCast library (phase 1 and 2). We hypothesized that by using a conditional generative adversarial network (cGAN) or deep neural networks (DNN), and leveraging this large set of toxicity data we could efficiently predict toxic outcomes of untested chemicals. Utilizing a novel method in this space, we converted the 3D structural information into a weighted set of points while retaining all information about the structure. *In vivo* toxicity and chemical data were used to train two neural network generators. The first was a DNN (Go-ZT) while the second utilized cGAN architecture (GAN-ZT) to train generators to produce toxicity data. Our results showed that Go-ZT significantly outperformed the cGAN, support vector machine, random forest and multilayer perceptron models in cross-validation, and when tested against an external test dataset. By combining both Go-ZT and GAN-ZT, our consensus model improved the SE, SP, PPV, and Kappa, to 71.4%, 95.9%, 71.4% and 0.673, respectively, resulting in an area under the receiver operating characteristic (AUROC) of 0.837. Considering their potential use as prescreening tools, these models could provide *in vivo* toxicity predictions and insight into the hundreds of thousands of untested chemicals to prioritize compounds for HT testing.

Support was also received from the Statistical and Applied Mathematical Sciences Institute (AJG, MJM, JD, MC). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

Author summary

A combined deep neural network (DNN) and conditional Generative Adversarial Network (cGAN) can leverage a large chemical set of experimental toxicity data plus chemical structure information to predict the toxicity of untested compounds.

Introduction

Currently, there are 85,000 chemicals registered with the EPA, as part of the Toxic Substances Control Act[1], that are manufactured, processed, or imported into the United States; however, only 4,400 have rigorous toxicological data, leaving over 80,000 chemicals untested [2,3]. Due to the high cost, and ethical concerns over the use of low-throughput mammalian models associated with traditional *in vitro* and *in vivo* assays, there has been increasing demand to reduce the number of animals used in toxicity testing paradigms by switching to *in silico* methods [4]. To directly address this chemical data gap and help prioritize chemicals for testing, both computational and high-throughput screening (HTS) approaches have been employed. The EPA developed the ToxCast program, an HTS approach, which included approximately 700 biochemical and cell-based assays, which was efficient but lacking in systemic biological complexity [2,5,6]. Therefore, as part of an effort to expand the toxicology database, a multidimensional HTS assay was devised to examine all ToxCast phase 1 and 2 chemicals (over 1,000 unique chemicals) for developmental- and neuro-toxicity in the embryonic zebrafish [7]. While computational approaches to bridge the data gap above have been developed, with Quantitative Structure-Activity Relationship (QSAR) and Read-Across being the most commonly used methodologies [8–13]. Both methods rely on the grouping of chemicals together using fragment descriptors, e.g. number of carbons, types of bonds, functional groups, etc. and have employed statistical or machine learning approaches [14–16]. Although these methods have been useful in identifying priority compounds for further testing, how these chemicals are grouped together might add bias, and recent machine learning advances have not been thoroughly explored [14,17].

Machine learning is a method of data analysis that automates the building of analytical models [18]. It is a branch of artificial intelligence based on the idea that systems can learn from data, identify patterns, and make decisions[19,20]. It encompasses a very broad range of supervised (minimal human intervention) and unsupervised (no human intervention) algorithms. Generally developed for computer science, sophisticated nonlinear machine learning algorithms have been increasingly used in cheminformatics and predictive toxicology with support vector machines (SVM), random forest (RF), deep neural networks (DNN), and Bayesian based methods being the most widely used [9,16,21–33]. More recently, GANs have gained prominence, where two neural networks (generator vs discriminator) are pitted against each other to generate a data distribution similar to the input [34,35]. This methodology was successfully used to design *de novo* molecules with desired properties in drug discovery and photovoltaic material design [36–39]. GANs can be extended to a conditional model (cGAN) if both the generator and discriminator are trained using some extra information, in this case a unique identifier.

Although GANs have been used to design new molecules, to our knowledge no research has been done to investigate their utility in predictive toxicology. Considering that tens of thousands of chemicals are manufactured or imported into the United States annually without rigorous toxicity data, it is imperative that new, structure-based models are developed to predict toxicity for priority testing. Therefore, the objective of this project is to use DNN and

cGAN to leverage the zebrafish HTS assay data along with chemical structure information to predict the toxic outcomes of untested chemicals.

Materials and methods

In this section, we describe a cGAN and DNN utilizing a novel 3D molecular vectorization algorithm to predict active developmental toxicants. An overview of our approach is shown in Figs 1 and 2. First, we used experimental data collected on a large, diverse compound set to assess the toxic effects of these chemicals following developmental exposure (Fig 3). Next, we recast the chemical data in a structural representation that maintained connectivity and positional information of each atom in the molecule but in a format easily read as input into a neural network. Next, we trained two types of generators to produce toxicity data using the recast chemical structural representation. The first used a deep neural network (DNN) with regression (Fig 1) while the second utilized a cGAN architecture for training (Fig 2). This was done to produce generators capable of predicting developmental zebrafish toxicity data dependent on chemical structure alone. Regression training was leveraged to maximize model fit, minimize training time, and utilized a simpler toxicity data representation. cGAN training minimized the effects of outliers and increased network adaptability to chemical structure. All feature layers and toxicity layers shown in Figs 1 and 2 are DNN's. These generators were trained using phase 1 and 2 ToxCast chemical data (n = 1003) split 80:20 into training and validation sets (Fig 3). Finally, we evaluated the trained networks on an independent test set containing chemicals (n = 56) of greater diversity in terms of both size and atomic constituents (Fig 3). These data were collected as part of an ongoing follow-up screen.

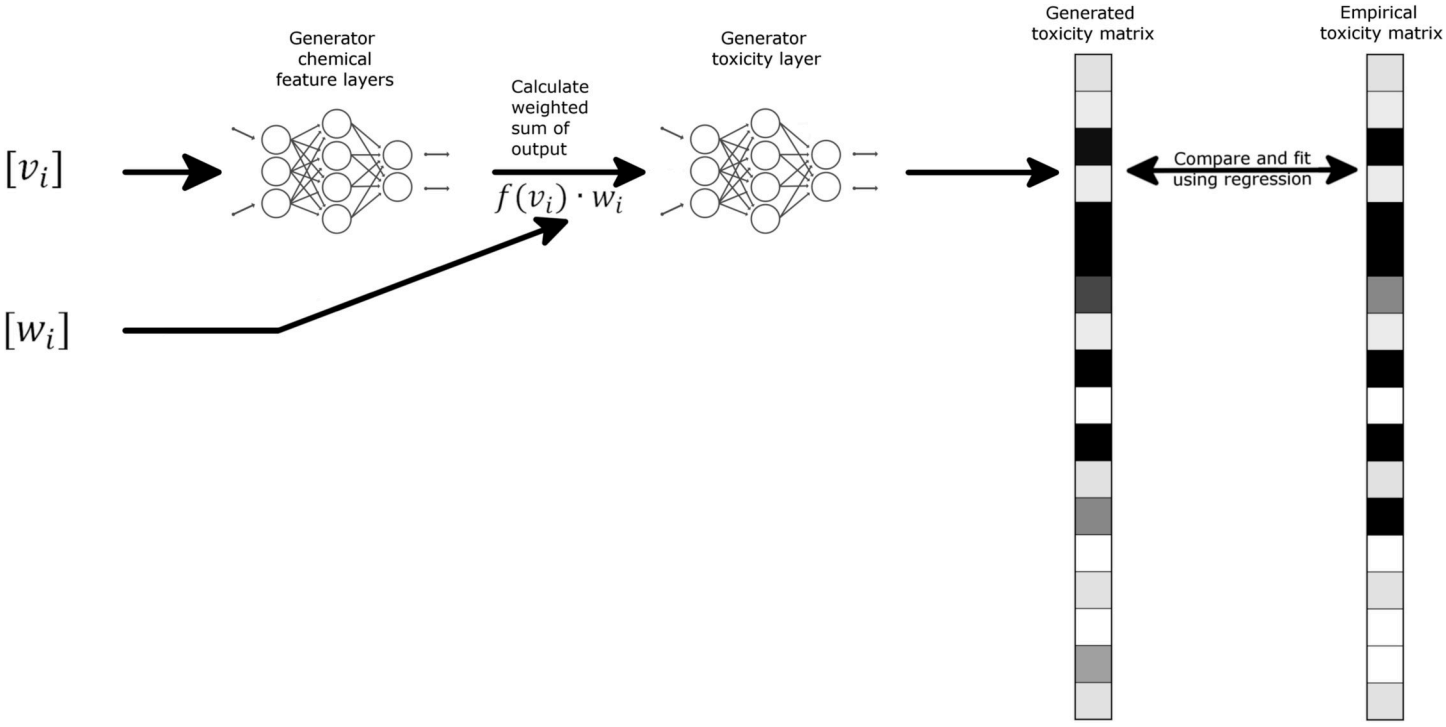

Fig 1. Regression generator diagram. Schematic representation of Go-ZT architecture showing chemical structural input represented as weights (w_i) and views (v_i) matrices passed through two fully connected neural networks to produce a predicted toxicity matrix. Darker matrix shading indicates higher toxicity values.

https://doi.org/10.1371/journal.pcbi.1009135.g001

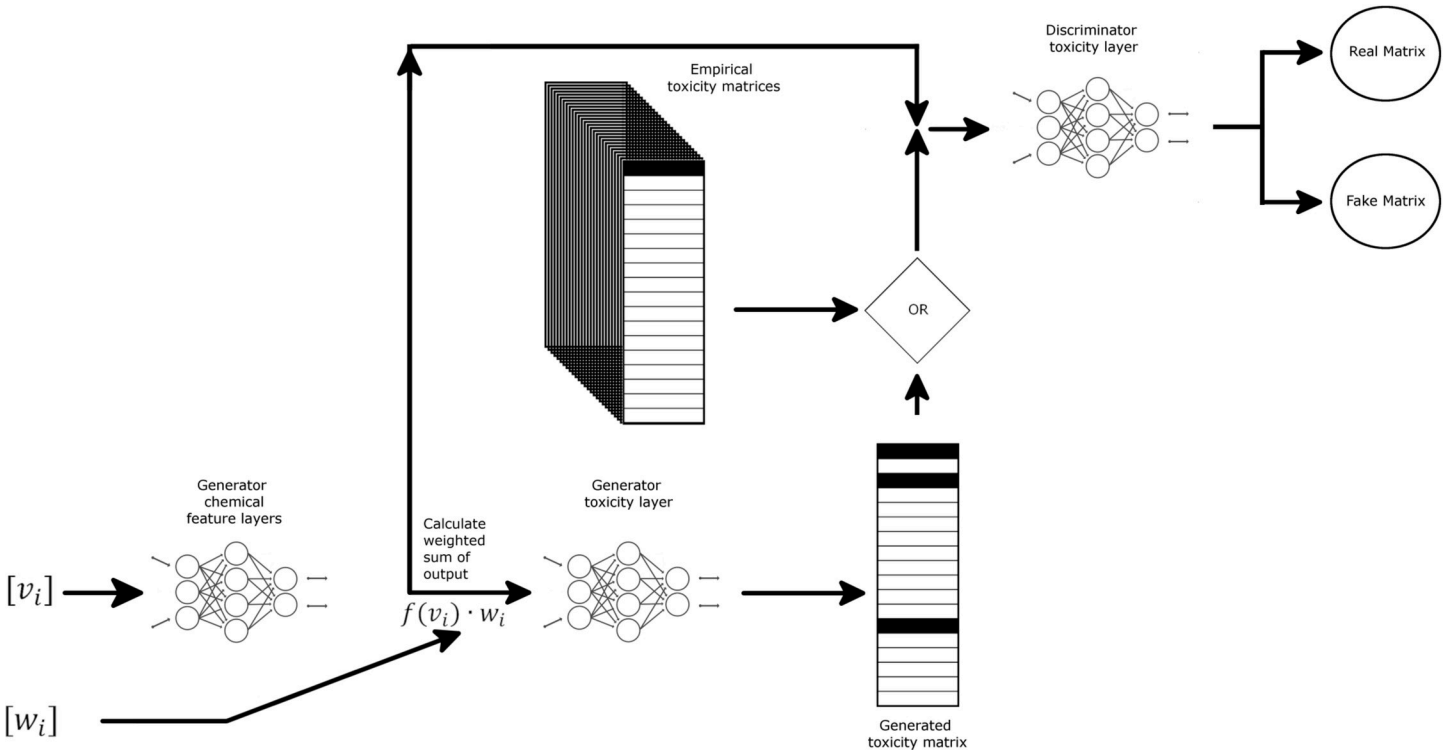

Fig 2. Conditional GAN diagram. Schematic representation of GAN-ZT architecture showing chemical structural input represented as weights (w_i) and views (v_i) matrices passed through two fully connected neural networks to produce a predicted toxicity matrix. Chemical features along with predicted or empirical toxicity matrices are then passed to a discriminator comprising a fully-connected neural network. Darker matrix shading indicates higher toxicity values.

https://doi.org/10.1371/journal.pcbi.1009135.g002

Empirical (experimental) data

The empirical data used to develop a generator of zebrafish toxicity were gathered as described in Truong et al. and Noyes et al.[7,40]. Fig 4 shows the experimental design. The data included

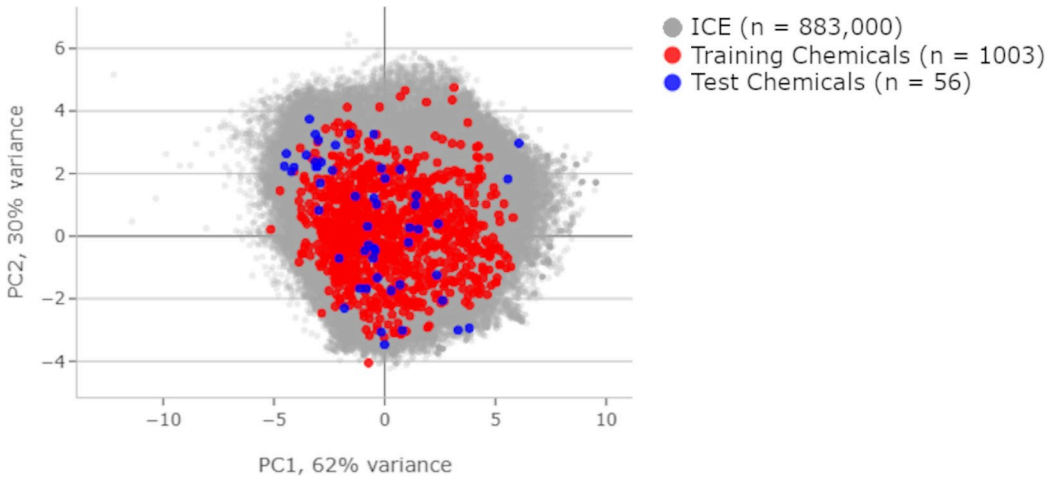

Fig 3. Data subdivision. Principal component analysis displayed against the background of over 800,000 chemicals in the Integrated Chemical Environment database. Compares physical chemical properties between the training and test sets.

https://doi.org/10.1371/journal.pcbi.1009135.g003

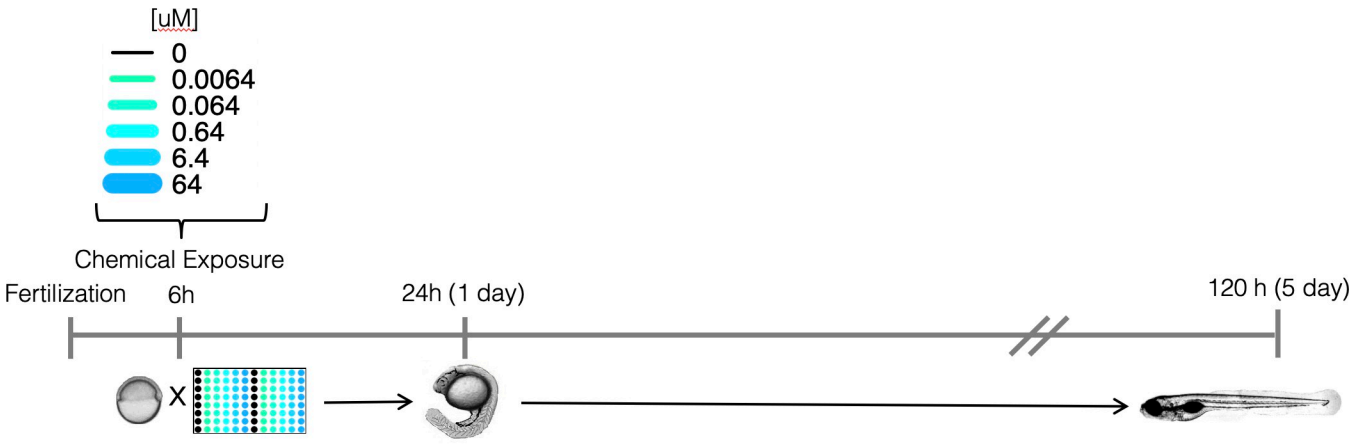

Fig 4. Experimental design. Schematic representation of the experimental approach for screening developmental and neurotoxicity of chemicals in larval zebrafish.

https://doi.org/10.1371/journal.pcbi.1009135.g004

1003 unique ToxCast chemicals tested at six concentrations for each chemical (0 μM, 0.0064 μM,0.064 μM, 0.64 μM, 6.4 μM and 64 μM). To minimize effects of response variability at lower concentrations, only the highest concentration was chosen for network training. There were 32 replicates (an individual embryo in singular wells of a 96-well plate) at each concentration for each chemical. At 120 hours post-fertilization (hpf), 18 distinct developmental endpoints were evaluated. The data were recorded as binary incidences and used to develop machine-learning models.

In a similar manner, toxicity matrices were created for an independent external test set of 56 chemicals that were collected in new experiments after collection of the original ToxCast data. This new test chemical set was more diverse in terms of atomic species and physical chemical properties (Fig 3)[41]. Due to chemical vectorization constraints we defined a reasonable domain of applicability to exclude, Perfluorinated chemicals with carbon chains longer than nine, Chloroperfluoro chemicals, and chemicals with a betaine functional group. Fig 3 shows the division of these data into train/test subsets and Principal Component Analysis (PCA) comparisons of the physical chemical properties using the Integrated Chemical Environment chemical characterization tool to highlight the diversity of the chemical domain space [41]. The PCA analysis includes the following physical chemical properties: Molecular Weight, Boiling Point, Henry's Law, Constant Melting Point, Negative Log of Acid Dissociation Constant, Octanol-Air Partition Coefficient, Octanol-Water Distribution Coefficient, Octanol-Water Partition Coefficient, Vapor Pressure, and Water Solubility[41].

Representing chemical compounds

The molecular structure of a chemical can be described or represented in various levels of complexity including molecular formula (1D), two-dimensional structural formula (2D), and three-dimensional, conformation-dependent (3D) with 2D being the most popular among chemists [42]. All three methods have been used to encode this structural information for utilization in deep learning, including, chemical properties, molecular fingerprints, SMILES, and graph vectorization, as well as 2D images of a chemical [38,43,44]. Considering that 2D representations are the most popular, ToxPrints, a molecular fingerprinting method will be used for benchmark evaluation. Utilizing CAS numbers chemical structural information and ToxPrints were retrieved from the EPA's Chemistry Dashboard [45]. The structural information was

converted from SDF to PDB format using Open Babel [46]. The PDB format was chosen as it is easily accessible and contains 3D structural information for all atoms in a molecule. Though a number of quantum chemistry based methods and software packages are available for 3D molecular vectorization [42], in this analysis, we utilized an novel algorithm developed to map and vectorize structure that was originally created for use in material sciences [47]. The PDB file for each chemical was vectorized as described by d'Avazac et al. [47] and illustrated in Fig 5. This method is simple and universal with few parameters and was adapted as follows: a view was started from each atom, or, by user option, only from each carbon atom (when available).

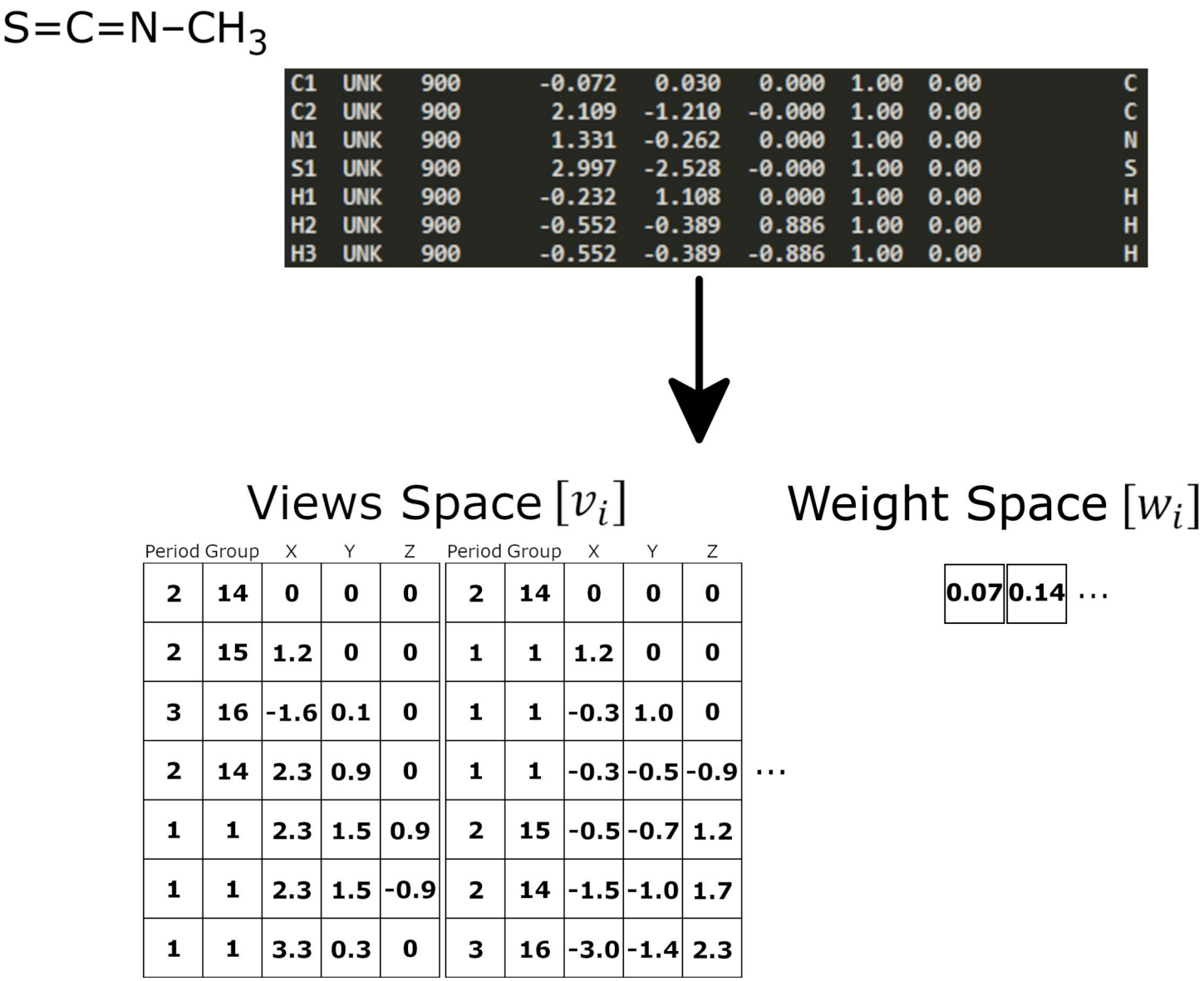

Fig 5. Diagram showing the vectorization of Methyl isothiocyanate. Atom information from the PDB file (shown in grey) in converted into the views and weights matrices. The views space (v_i) columns one and two identify the chemical species and correspond to an atom's position on the periodic table indicating their period and group, respectively. While the last three columns show the relative position of each atom. The weight space (w_i) values correspond to each of the views space matrices. In the first views Table C1 is set at the center while in the second view C2 is set at the center of the view. This molecule has nine views, which can be reduced to three views if preference is given only to carbon.

https://doi.org/10.1371/journal.pcbi.1009135.g005

Onto each view, the remaining atoms (up to a user-defined limit) were added in order of their distance from the first atom; ties were broken by spitting a view into two or more views and the origin and orientation were determined by canonical rules. Lastly, an atom's position on the periodic table was used as a unique identifier (group and period), together with location (x, y, and z coordinates) producing five chemical features per atom.

Network performance and evaluation

Previous work analyzing this data has shown that a summary value, aggregate entropy (AggE) can be calculated from the 18 morphological endpoints[48]. The intent of this summary value is to capture a meaningful measure of toxicity, while avoiding overinflation by summing highly correlated endpoints. Using the threshold value (9.35) identified by Zhang et al. for AggE in these data, compounds may be classified as active or inactive[48]. It should be noted that this threshold value influences the toxicity hit rate of a chemical and is concentration dependent. Therefore, it would need to be changed when investigating other nominal concentrations. Following training, the resulting generators were used to output toxicity matrices. AggE values were calculated using both the empirical and generated toxicity matrices and compounds were classified as active or inactive using the threshold value identified above. Active vs inactive classification accuracy was evaluated using a confusion matrix, Cohen's Kappa statistic, and area under the receiver operating characteristic (AUROC) as Kappa and AUROC measure model accuracy, while compensating for simple chance[49]. The primary metrics we used from the confusion matrix included sensitivity (SE), specificity (SP), and positive predictive value (PPV) as these parameters give us the true positive rate, true negative rate, and the proportion of true positives amongst all positive calls[50–52]. The network with the highest Kappa statistic and positive predictive value (PPV) was used for evaluation of the test dataset.

Data imbalance

All datasets showed strong active vs inactive class imbalance (Table 1). Classifiers may be biased towards the major class (inactive) and, therefore, show poor performance accuracy for the minor class (active) [53]. To address this problem, we used Cohens Kappa statistic and positive predictive value (PPV) to evaluate model performance.

cGAN and regression generator

Two network architectures were developed and tested to train a generator that was capable of using 3D chemical information, in the form of vectorize views (Fig 5), and generate a toxicity matrix (Figs 1 and 2). The following two different models were trained on a Dell R740 containing two Intel Xeon processors with 18 cores per processor, 512 GB RAM, and a Tesla-V100-PCIE (31.7 GB) using the open source Python library Keras [54] on top of TensorFlow as the backend [55] within a purpose build Singularity container environment [56]. Swish activation for hidden and output layers, batch normalization between layers, mean squared error for the loss function, stochastic gradient descent (SGD) as the generator optimizer, Adam as the discriminator optimizer, a kernel initializer with a Gaussian distribution, and without dropout were used [57–60]. For each model, we optimized the hyperparameters (i.e, the

Table 1. Summary of training and testing data used in this study.

Data	Active Chemical	Inactive Chemical	Total
Training data	159	844	1003
External testing data	7	49	56

https://doi.org/10.1371/journal.pcbi.1009135.t001

number of hidden layers, the number of nodes in the layers, the number of chemical views, number of atoms per view, loss functions, optimizers, learning rates etc.) by random search technique followed by a 10-fold cross-validation using a randomized 80:20 split using Cohens Kappa statistic as the objective metric.

The first model was a simpler deep neural network trained to produce a toxicity matrix using regression (Fig 1). We used multiple layers consisting of a deep neural network base layer to extract salient features from the views matrix for each chemical structure (generator chemical feature layer). A second layer calculated the weighted sum over features ($f(v_i)\cdot w_i$) and a final deep neural network (generator toxicity layer) generates toxicity values. The regression generator (Go-ZT) was trained over the course of 75 epochs (7 seconds/fold).

The second model developed (GAN-ZT) used a much more complex cGAN architecture to generate a toxicity matrix (Fig 2). Similar to the Go-ZT above, we used multiple layers consisting of a deep neural network base layer to extract salient features from the views matrix for each chemical structure (generator chemical feature layer) and a second layer to calculate the weighted sum over these features ($f(v_i)\cdot w_i$). The resulting weighted sum of views ($f(v_i)\cdot w_i$) for each chemical was used as input to a final deep neural network (generator toxicity layer) to produce a generated toxicity matrix. The discriminator took the generators resulting weighted sum of views ($f(v_i)\cdot w_i$) for each chemical along with its corresponding empirical or generated toxicity matrices to determine whether the toxicity matrix was real or fake. This information was then backpropagated to train the generator. GAN-ZT was trained over the course of 2000 epochs (2 hours/fold). Fig 6 shows the training loss for both Go-ZT and GAN-ZT.

Other classifiers

To evaluate the performance of the deep-learning and cGAN models in relation to other methods and chemical representation, we constructed the support vector machines (SVM), multilayer perceptron (MLP), and random forest (RF) models using KNIME (version 4.3.2) [61], and EPA ToxPrints, respectively [45]. We optimized the hyperparameters by a randomized search technique with cross-validation, using Cohens Kappa as the objective metric.

Results

Performance of five machine learning algorithms using cross-validation

Empirical and generated toxicity matrices for each chemical from the training and test datasets (Fig 3) were used to calculate AggE values to determine activity classification. The empirical

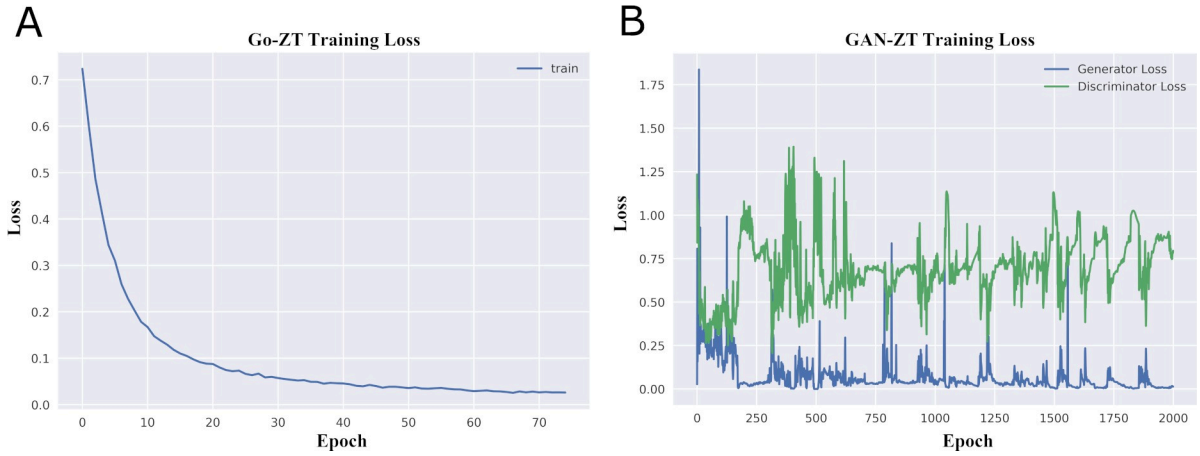

Fig 6. Go-ZT and GAN-ZT loss functions during training. Changes of loss functions during the training of (A) Go-ZT and (B) GAN-ZT.

https://doi.org/10.1371/journal.pcbi.1009135.g006

Table 2. Performance of different methods in activity classification with 10-fold cross-validation.

Model	SE		SP		PPV		Kappa		AUROC	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
SVM	17.7	10.9	92.1	3.6	0.30	15.7	0.115	0.13	0.410	0.10
MLP	12.2	7.60	94.5	1.6	28.2	14.3	0.085	0.10	0.607	0.07
RF	6.5	6.10	**98.2**	3.1	56.7	44.1	0.071	0.09	0.609	0.08
GAN-ZT	**58.4**	20.7	64.1	19.4	28.4	5.4	0.160	0.05	0.613	0.03
Go-ZT	44.6	7.25	97.1	1.65	**76.1**	10.2	**0.495**	0.08	**0.709**	0.04

https://doi.org/10.1371/journal.pcbi.1009135.t002

training dataset contains 159 chemicals that meet the AggE threshold of 9.35 to be classified as an active compound (15.9%), as shown in Table 1. Table 2 shows the mean and standard deviation for the five machine learning algorithms trained using a 10-fold cross-validation. Go-ZT and GAN-ZT outperformed SVM, MLP, and RF using Kappa as the primary measure of performance. The GAN-ZT, SVM, MLP, and RF models showed poor performance while Go-ZT achieved moderate predictive performance and a good PPV percentage.

External validation using independent test data

We built final models using all training data and their best parameters, following hyperparameter optimization, and assessed their performance using an external testing dataset containing 7 active compounds (12.5%) as shown in Table 1. Considering the very slow pace of GAN-ZT training we used the same number of chemical features, number of views, number of atoms per view, and number of hidden layers for the generator chemical feature layers (three), and generator toxicity layers (11) for both GAN-ZT and Go-ZT (Fig 1 and 2). As shown in Table 3, four of the five models showed improved performance on the test set with GAN-ZT showing a slight decline. Once again Go-ZT outperformed all other models. Go-ZT produced an SE, SP, and PPV of 71.4%, 91.8%, and 55.6% respectively. GAN-ZT on the other hand produced SE, SP, and PPV values of 71.4%, 59.2%, and 20.0%, respectively. Evaluation of the chemical domain space using Go-ZT and GAN-ZT showed that chemicals should be excluded due to long chain length, and betaine or Chloroperfluoro functional groups as these chemical properties fall outside of the domain space of our models. The results show that Go-ZT performed best with increases in SE, PPV, Kappa, and AUROC values while GAN-ZT saw declines in PPV, and Kappa values (Fig 7).

Combined model

Model combinations between Go-ZT, and GAN-ZT, SVM, MLP, or RF were assessed for improvement with particular focus on PPV, Kappa, and AUROC. By combining the predictive results of Go-ZT and GAN-ZT we were able to improve the SP, Kappa, and AUROC to 95.9%, 0.673, and 0.837, respectively (Fig 8). As a result of the consensus between the models we were

Table 3. Performances of different methods in activity prediction of test set chemicals.

Model	SE	SP	PPV	Kappa	AUROC
SVM	28.6	95.9	50.0	0.300	0.649
MLP	28.6	89.8	28.6	0.184	0.660
RF	28.6	98.0	66.7	0.351	0.459
GAN-ZT	71.4	59.2	20.0	0.146	0.653
Go-ZT	71.4	91.8	55.6	**0.564**	**0.816**

https://doi.org/10.1371/journal.pcbi.1009135.t003

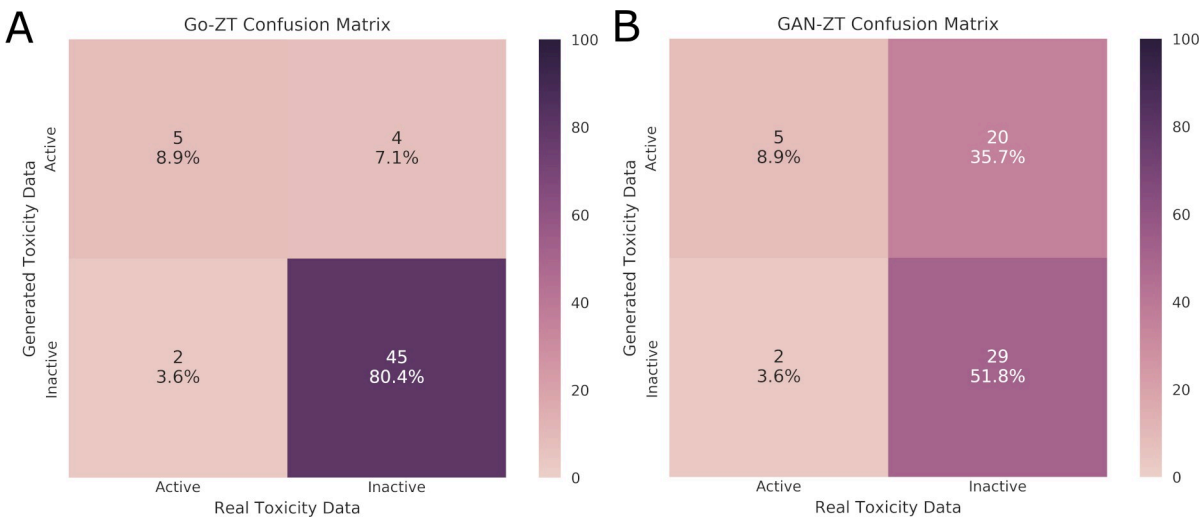

Fig 7. Test dataset confusion matrices. Evaluation of the classification of chemicals in the test data set as either active or inactive using real versus generated toxicity matrices by (A) Go-ZT or (B) GAN-ZT. Color scale represents percent of total chemicals.

https://doi.org/10.1371/journal.pcbi.1009135.g007

able to capture five of the seven active chemicals in the test set while eliminating two false positives which translates to a PPV of 71.4%.

Random label shuffling shows no predictive power

We performed 1000 random shuffling's to construct 1000 label shuffled test datasets for analysis using our models. Table 4 shows that the models performed poorly. The poor performance indicated that random noise is unlikely driving model performance.

Discussion

GAN-ZT and Go-ZT architecture with chemical structure vectorization using views predicts empirical toxicity results with fair to good Kappa values of 0.160 and 0.495, respectively. GAN-ZT, SVM, RF, and MLP models performed similarly on the training dataset while the SVM and RF performed better than the GAN-ZT and MLP on the test dataset. Go-ZT significantly outperformed all models on both the training and test datasets. Go-ZT predicted active chemicals with an SE of 71.4%, SP of 91.8%, and a PPV of 55.6% while GAN -ZT predicted active chemicals with an SE of 71.4%, SP of 59.2%, and a PPV of 20.0%. When we examined the overlap in predicted active chemicals between Go-ZT and the other four models only in combination with GAN-ZT did we see that a consensus model improved the SP, PPV, Kappa, and AUROC to 95.9%, 71.4%, 0.673, and 0.837, respectively. These results show that a regression-based DNN is capable of predicting toxic developmental activity with good efficacy in both the training and test datasets. Further, by leveraging the strengths of both the supervised generative adversarial network and DNN the intersection between the models was able to accurately predict the toxicity of chemicals not part of the initial ToxCast screen.

A wide range of machine learning methods such as Deep Neural Networks (DNN), Support Vector Machines (SVM), *k*-nearest neighbor (*k*-NN), gradient-boosted decision trees, and Bayesian Classifiers have been applied to cheminformatics problems to predict biologically active chemicals with AUROC values ranging from 0.7–0.83 [62]. These studies utilized molecular fingerprints of chemicals from the ChEMBL database, and single biological activity prediction in drug discovery but were not focused on *in vivo* toxicity. More recent studies have

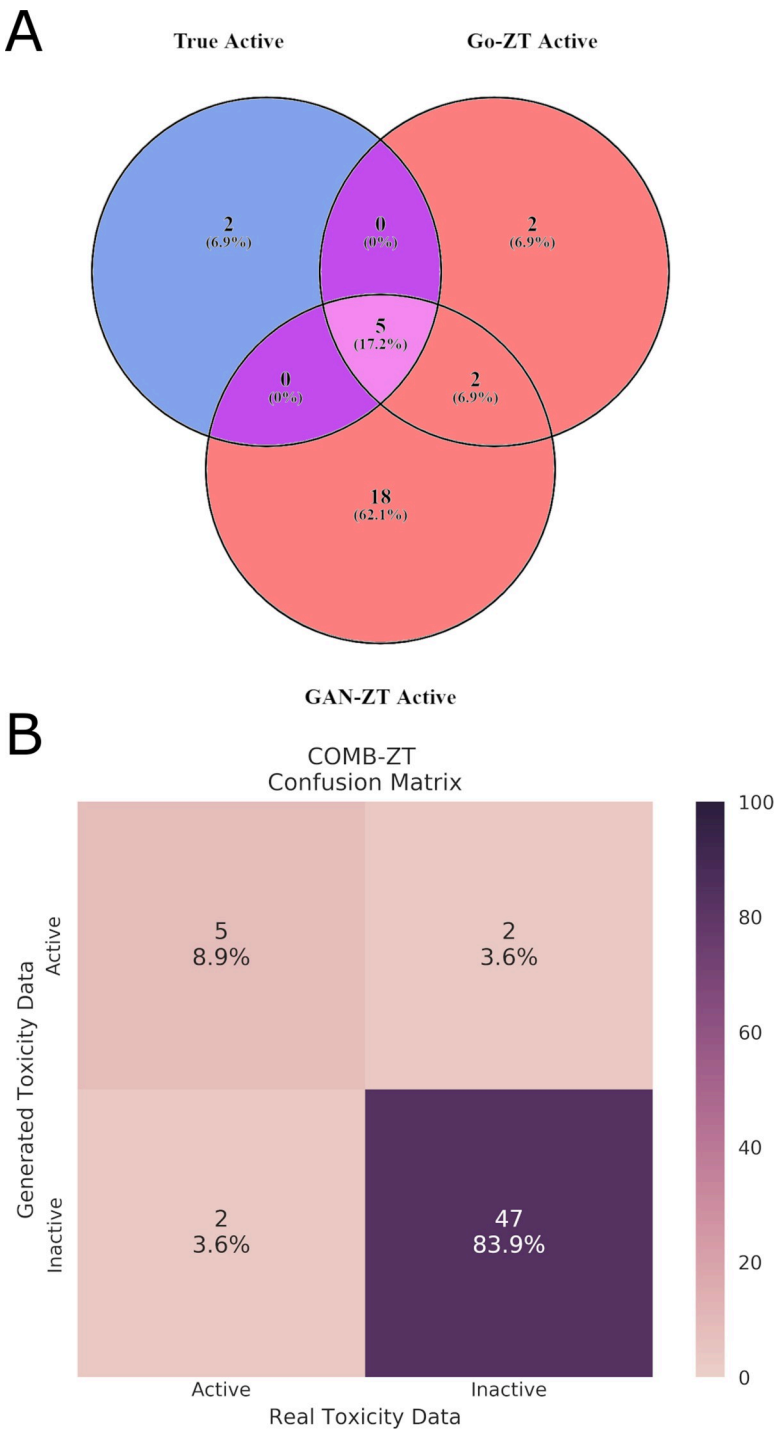

Fig 8. Model consensus on chemical activity. (A)Venn diagram showing the overlap between true active chemicals and chemicals predicted to be active by either Go-ZT or GAN-ZT. (B) A confusion matrix showing the performance of the combined Go-ZT and GAN-ZT models using the test dataset.

https://doi.org/10.1371/journal.pcbi.1009135.g008

used data from the Tox21 database as part of the NIH 2014 Tox21 Data Challenge with multi-task DNNs outperforming other machine learning methods with AUROC values ranging from 0.69–0.92 in 12 different biochemical assays [44]. Further, Mansouri and Judson successfully

Table 4. Model performance using shuffled data.

Model	Kappa		AUROC	
	Mean	SD	Mean	SD
GAN-ZT	0.004	0.10	0.504	0.10
Go-ZT	0.021	0.13	0.513	0.08

https://doi.org/10.1371/journal.pcbi.1009135.t004

built a QSARs model for G-protein coupled receptor assays using partial least square discriminant analysis that resulted in a balanced accuracy of 96%[63]. Our combined model produced a similar AUROC and a lower balanced accuracy value (83.7%). To the best of our knowledge, this is the first study to develop a DNN model without explicit use of molecular descriptors to predict *in vivo* toxicity in a large chemical set. While our model is a potentially useful tool for prioritizing chemicals for screening tests, it does have its limitations including chemical domain space, and our networks are dependent on accurate 3D chemical structural information to produce reliable results and at this time are not designed to evaluate mixtures.

Overall, our results show that a DNN utilizing 3D chemical structural information is a useful prescreening tool for predicting the toxic outcomes of the approximately 80,000 untested chemicals registered with the EPA. If we consider that between the training, and test datasets there are 1,059 chemicals and only 166 are active (15.7%) then there are possibly 12,540 untested active chemicals registered. If the PPV of the combined model holds at 71.4% this would result in a list of ~17,500 chemicals to screen. While still a very large number it would reduce the experimental space by over three-quarters. Further, these compounds may then be ranked by AggE and the chemicals with the highest ranking identified should then be prioritized for assessment using the zebrafish HTS assay for developmental toxicity, as these assays are considerably faster and cheaper than traditional chemical screens in mammalian systems.

Looking to the future, increasing computational resources and chemical structural data, alternative network architectures, inclusion of ToxCast assay results, and zebrafish behavioral endpoints in conditional training could improve the predictive value of DNN in *in vivo* toxicity testing. The views chemical vectorization methodology needs to be further evaluated with existing machine learning algorithms. There is also potential to add other chemical information to the views methodology including charge and types of bonds. Additional work needs to be done to assess the utility of cGANs as a tool to evaluate structure activity relationships (SAR) in *in vitro* toxicology and finally DNNs need to be adapted to evaluate mixtures if a sufficiently large dataset is available.

Author Contributions

Conceptualization: Adrian J. Green.

Data curation: Lisa Truong.

Formal analysis: Adrian J. Green, Martin J. Mohlenkamp, Jhuma Das, Meenal Chaudhari, Lisa Truong, David M. Reif.

Funding acquisition: Robyn L. Tanguay, David M. Reif.

Investigation: Lisa Truong.

Methodology: Adrian J. Green, Martin J. Mohlenkamp, Jhuma Das, Meenal Chaudhari, David M. Reif.

Project administration: Adrian J. Green, Lisa Truong, Robyn L. Tanguay, David M. Reif.

Software: Adrian J. Green, Martin J. Mohlenkamp, Jhuma Das.

Supervision: Martin J. Mohlenkamp, Robyn L. Tanguay, David M. Reif.

Visualization: Adrian J. Green.

Writing – original draft: Adrian J. Green, Martin J. Mohlenkamp, Robyn L. Tanguay.

Writing – review & editing: Adrian J. Green, Martin J. Mohlenkamp, Jhuma Das, Meenal Chaudhari, Lisa Truong, Robyn L. Tanguay, David M. Reif.

References

1. US EPA O. About the TSCA Chemical Substance Inventory. In: US EPA [Internet]. 2 Mar 2015 [cited 23 Aug 2019]. Available: https://www.epa.gov/tsca-inventory/about-tsca-chemical-substance-inventory
2. US EPA O. ToxCast Chemicals. In: US EPA [Internet]. 25 Oct 2017 [cited 23 Aug 2019]. Available: https://www.epa.gov/chemical-research/toxcast-chemicals
3. Richard AM, Judson RS, Houck KA, Grulke CM, Volarath P, Thillainadarajah I, et al. ToxCast Chemical Landscape: Paving the Road to 21st Century Toxicology. Chem Res Toxicol. 2016; 29: 1225–1251. https://doi.org/10.1021/acs.chemrestox.6b00135 PMID: 27367298
4. Krewski D, Acosta D, Andersen M, Anderson H, Bailar JC, Boekelheide K, et al. TOXICITY TESTING IN THE 21ST CENTURY: A VISION AND A STRATEGY. J Toxicol Environ Health B Crit Rev. 2010; 13: 51–138. https://doi.org/10.1080/10937404.2010.483176 PMID: 20574894
5. Judson RS, Houck KA, Kavlock RJ, Knudsen TB, Martin MT, Mortensen HM, et al. In vitro screening of environmental chemicals for targeted testing prioritization: the ToxCast project. Environ Health Perspect. 2010; 118: 485–492. https://doi.org/10.1289/ehp.0901392 PMID: 20368123
6. Dix DJ, Houck KA, Martin MT, Richard AM, Setzer RW, Kavlock RJ. The ToxCast program for prioritizing toxicity testing of environmental chemicals. Toxicol Sci. 2007; 95: 5–12. https://doi.org/10.1093/toxsci/kfl103 PMID: 16963515
7. Truong L, Reif DM, St Mary L, Geier MC, Truong HD, Tanguay RL. Multidimensional In Vivo Hazard Assessment Using Zebrafish. Toxicol Sci. 2014; 137: 212–233. https://doi.org/10.1093/toxsci/kft235 PMID: 24136191
8. Matsuzaka Y, Uesawa Y. DeepSnap-Deep Learning Approach Predicts Progesterone Receptor Antagonist Activity With High Performance. Front Bioeng Biotechnol. 2020; 7. https://doi.org/10.3389/fbioe.2019.00485 PMID: 32039185
9. Idakwo G, Thangapandian S, Luttrell JI, Zhou Z, Zhang C, Gong P. Deep Learning-Based Structure-Activity Relationship Modeling for Multi-Category Toxicity Classification: A Case Study of 10K Tox21 Chemicals With High-Throughput Cell-Based Androgen Receptor Bioassay Data. Front Physiol. 2019; 10. https://doi.org/10.3389/fphys.2019.01044 PMID: 31456700
10. Pawar G, Madden JC, Ebbrell D, Firman JW, Cronin MTD. In Silico Toxicology Data Resources to Support Read-Across and (Q)SAR. Front Pharmacol. 2019; 10. https://doi.org/10.3389/fphar.2019.00561 PMID: 31244651
11. Yoo JW, Kruhlak NL, Landry C, Cross KP, Sedykh A, Stavitskaya L. Development of improved QSAR models for predicting the outcome of the in vivo micronucleus genetic toxicity assay. Regulatory Toxicology and Pharmacology. 2020; 113: 104620. https://doi.org/10.1016/j.yrtph.2020.104620 PMID: 32092371
12. Ghorbanzadeh M, Zhang J, Andersson PL. Binary classification model to predict developmental toxicity of industrial chemicals in zebrafish. Journal of Chemometrics. 2016; 30: 298–307. https://doi.org/10.1002/cem.2791
13. Zhang H, Ren J-X, Kang Y-L, Bo P, Liang J-Y, Ding L, et al. Development of novel in silico model for developmental toxicity assessment by using naïve Bayes classifier method. Reproductive Toxicology. 2017; 71: 8–15. https://doi.org/10.1016/j.reprotox.2017.04.005 PMID: 28428071
14. Baskin II. Machine Learning Methods in Computational Toxicology. In: Nicolotti O, editor. Computational Toxicology: Methods and Protocols. New York, NY: Springer; 2018. pp. 119–139. https://doi.org/10.1007/978-1-4939-7899-1_5
15. Yang C, Tarkhov A, Marusczyk J, Bienfait B, Gasteiger J, Kleinoeder T, et al. New Publicly Available Chemical Query Language, CSRML, To Support Chemotype Representations for Application to Data Mining and Modeling. J Chem Inf Model. 2015; 55: 510–528. https://doi.org/10.1021/ci500667v PMID: 25647539

16. Mitchell JBO. Machine learning methods in chemoinformatics. WIREs Computational Molecular Science. 2014; 4: 468–481. https://doi.org/10.1002/wcms.1183 PMID: 25285160

17. Non-test Methods (Q)SAR and Read-across. In: AltTox.org [Internet]. 3 Nov 2014 [cited 23 Aug 2019]. Available: http://alttox.org/mapp/emerging-technologies/non-test-approaches-qsars-read-across/

18. Machine Learning: a Method of Data Analysis that Automates Analytical Model Building. Data Analytics and Big Data. John Wiley & Sons, Ltd; 2018. pp. 101–122. https://doi.org/10.1002/9781119528043.ch6

19. Machine Learning: What it is and why it matters. [cited 12 Dec 2018]. Available: https://www.sas.com/en_us/insights/analytics/machine-learning.html

20. What is Machine Learning? 25 Mar 2021 [cited 28 Apr 2021]. Available: https://www.ibm.com/cloud/learn/machine-learning

21. Ekins S. Progress in computational toxicology. Journal of Pharmacological and Toxicological Methods. 2014; 69: 115–140. https://doi.org/10.1016/j.vascn.2013.12.003 PMID: 24361690

22. Ekins S, Puhl AC, Zorn KM, Lane TR, Russo DP, Klein JJ, et al. Exploiting machine learning for end-to-end drug discovery and development. Nature Materials. 2019; 18: 435–441. https://doi.org/10.1038/s41563-019-0338-z PMID: 31000803

23. Hu Q, Feng M, Lai L, Pei J. Prediction of Drug-Likeness Using Deep Autoencoder Neural Networks. Front Genet. 2018; 9. https://doi.org/10.3389/fgene.2018.00585 PMID: 30538725

24. Mayr A, Klambauer G, Unterthiner T, Steijaert M, Wegner JK, Ceulemans H, et al. Large-scale comparison of machine learning methods for drug target prediction on ChEMBL †Electronic supplementary information (ESI) available: Overview, Data Collection and Clustering, Methods, Results, Appendix. Chem Sci. 2018; 9: 5441–5451. See https://doi.org/10.1039/c8sc00148k PMID: 30155234

25. Pu L, Naderi M, Liu T, Hsiao-Chun W, Mukhopadhyay S, Brylinski M. eToxPred: a machine learning-based approach to estimate the toxicity of drug candidates. BMC Pharmacology & Toxicology. 2019; 20. https://doi.org/10.1186/s40360-018-0282-6 PMID: 30621790

26. Wang H, Liu R, Schyman P, Wallqvist A. Deep Neural Network Models for Predicting Chemically Induced Liver Toxicity Endpoints From Transcriptomic Responses. Front Pharmacol. 2019; 10. https://doi.org/10.3389/fphar.2019.00042 PMID: 30804783

27. Yuan Q, Wei Z, Guan X, Jiang M, Wang S, Zhang S, et al. Toxicity Prediction Method Based on Multi-Channel Convolutional Neural Network. Molecules. 2019; 24: 3383. https://doi.org/10.3390/molecules24183383 PMID: 31533341

28. Webel HE, Kimber TB, Silke R, Neuenschwander M, Marc N, Volkamer A. Revealing cytotoxic substructures in molecules using deep learning. Journal of Computer—Aided Molecular Design. 2020; 34: 731–746. https://doi.org/10.1007/s10822-020-00310-4 PMID: 32297073

29. Alizadeh R, Allen JK, Mistree F. Managing computational complexity using surrogate models: a critical review. Res Eng Design. 2020; 31: 275–298. https://doi.org/10.1007/s00163-020-00336-7

30. Ho TK. The random subspace method for constructing decision forests. IEEE Transactions on Pattern Analysis and Machine Intelligence. 1998; 20: 832–844. https://doi.org/10.1109/34.709601

31. Cortes C, Vapnik V. Support-vector networks. Mach Learn. 1995; 20: 273–297. https://doi.org/10.1007/BF00994018

32. Idakwo G, Luttrell J, Chen M, Hong H, Zhou Z, Gong P, et al. A review on machine learning methods for in silico toxicity prediction. Journal of Environmental Science and Health, Part C. 2018; 36: 169–191. https://doi.org/10.1080/10590501.2018.1537118 PMID: 30628866

33. Idakwo G, Thangapandian S, Luttrell J, Li Y, Wang N, Zhou Z, et al. Structure–activity relationship-based chemical classification of highly imbalanced Tox21 datasets. Journal of Cheminformatics. 2020; 12: 66. https://doi.org/10.1186/s13321-020-00468-x PMID: 33372637

34. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, et al. Generative Adversarial Nets. In: Ghahramani Z, Welling M, Cortes C, Lawrence ND, Weinberger KQ, editors. Advances in Neural Information Processing Systems 27. Curran Associates, Inc.; 2014. pp. 2672–2680. Available: http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf

35. Goodfellow I. NIPS 2016 Tutorial: Generative Adversarial Networks. arXiv:170100160 [cs]. 2016 [cited 24 Sep 2019]. Available: http://arxiv.org/abs/1701.00160

36. Kadurin A, Nikolenko S, Khrabrov K, Aliper A, Zhavoronkov A. druGAN: An Advanced Generative Adversarial Autoencoder Model for de Novo Generation of New Molecules with Desired Molecular Properties in Silico. Mol Pharmaceutics. 2017; 14: 3098–3104. https://doi.org/10.1021/acs.molpharmaceut.7b00346 PMID: 28703000

37. Sanchez-Lengeling B, Outeiral C, Guimaraes GL, Aspuru-Guzik A. Optimizing distributions over molecular space. An Objective-Reinforced Generative Adversarial Network for Inverse-design Chemistry (ORGANIC). 2017. https://doi.org/10.26434/chemrxiv.5309668.v3

38. Guimaraes GL, Sanchez-Lengeling B, Outeiral C, Farias PLC, Aspuru-Guzik A. Objective-Reinforced Generative Adversarial Networks (ORGAN) for Sequence Generation Models. arXiv:170510843 [cs, stat]. 2018 [cited 14 Apr 2020]. Available: http://arxiv.org/abs/1705.10843

39. Putin E, Asadulaev A, Ivanenkov Y, Aladinskiy V, Sanchez-Lengeling B, Aspuru-Guzik A, et al. Reinforced Adversarial Neural Computer for de Novo Molecular Design. J Chem Inf Model. 2018; 58: 1194–1204. https://doi.org/10.1021/acs.jcim.7b00690 PMID: 29762023

40. Noyes PD, Haggard DE, Gonnerman GD, Tanguay RL. Advanced Morphological—Behavioral Test Platform Reveals Neurodevelopmental Defects in Embryonic Zebrafish Exposed to Comprehensive Suite of Halogenated and Organophosphate Flame Retardants. Toxicol Sci. 2015; 145: 177–195. https://doi.org/10.1093/toxsci/kfv044 PMID: 25711236

41. National Toxicology Program. ICE Tools. 21 Feb 2020 [cited 4 Aug 2020]. Available: https://ice.ntp.niehs.nih.gov/Tools

42. Cherkasov A, Muratov EN, Fourches D, Varnek A, Baskin II, Cronin M, et al. QSAR Modeling: Where have you been? Where are you going to? J Med Chem. 2014; 57: 4977–5010. https://doi.org/10.1021/jm4004285 PMID: 24351051

43. Hirohara M, Saito Y, Koda Y, Sato K, Sakakibara Y. Convolutional neural network based on SMILES representation of compounds for detecting chemical motif. BMC Bioinformatics. 2018; 19: 526. https://doi.org/10.1186/s12859-018-2523-5 PMID: 30598075

44. Mayr A, Klambauer G, Unterthiner T, Hochreiter S. DeepTox: Toxicity Prediction using Deep Learning. Front Environ Sci. 2016; 3. https://doi.org/10.3389/fenvs.2015.00080

45. Williams AJ, Grulke CM, Edwards J, McEachran AD, Mansouri K, Baker NC, et al. The CompTox Chemistry Dashboard: a community data resource for environmental chemistry. Journal of Cheminformatics. 2017; 9: 61. https://doi.org/10.1186/s13321-017-0247-6 PMID: 29185060

46. O'Boyle NM, Banck M, James CA, Morley C, Vandermeersch T, Hutchison GR. Open Babel: An open chemical toolbox. Journal of Cheminformatics. 2011; 3: 33. https://doi.org/10.1186/1758-2946-3-33 PMID: 21982300

47. d'Avezac M, Botts R, Mohlenkamp MJ, Zunger A. Learning to Predict Physical Properties using Sums of Separable Functions. SIAM J Sci Comput. 2011; 33: 3381–3401. https://doi.org/10.1137/100805959

48. Zhang G, Marvel S, Truong L, Tanguay RL, Reif DM. Aggregate entropy scoring for quantifying activity across endpoints with irregular correlation structure. Reprod Toxicol. 2016; 62: 92–99. https://doi.org/10.1016/j.reprotox.2016.04.012 PMID: 27132190

49. Ben-David A. About the relationship between ROC curves and Cohen's kappa. Engineering Applications of Artificial Intelligence. 2008; 21: 874–882. https://doi.org/10.1016/j.engappai.2007.09.009

50. Pearson K. On the theory of contingency and its relation to association and normal correlation. Drapers Company Research Memoirs. Dulau and Co.; 1904. Available: https://archive.org/details/cu31924003064833/page/n1/mode/2up

51. Townsend JT. Theoretical analysis of an alphabetic confusion matrix. Perception & Psychophysics. 1971; 9: 40–50. https://doi.org/10.3758/BF03213026

52. Parikh R, Mathai A, Parikh S, Chandra Sekhar G, Thomas R. Understanding and using sensitivity, specificity and predictive values. Indian J Ophthalmol. 2008; 56: 45–50. https://doi.org/10.4103/0301-4738.37595 PMID: 18158403

53. Lemaître G, Nogueira F, Aridas CK. Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning. Journal of Machine Learning Research. 2017; 18: 1–5.

54. Chollet F, others. Keras. GitHub; 2015. Available: https://github.com/fchollet/keras

55. Abadi Martín, Agarwal Ashish, Barham Paul, Brevdo Eugene, Chen Zhifeng, Citro Craig, et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. 2015. Available: https://www.tensorflow.org/

56. Sylabs.io. Singularity. Sylabs.io; 2019. Available: https://sylabs.io/singularity/

57. Kingma DP, Ba J. Adam: A Method for Stochastic Optimization. arXiv:14126980 [cs]. 2017 [cited 4 Sep 2020]. Available: http://arxiv.org/abs/1412.6980

58. Ramachandran P, Zoph B, Le QV. Searching for Activation Functions. arXiv:171005941 [cs]. 2017 [cited 4 Sep 2020]. Available: http://arxiv.org/abs/1710.05941

59. Osl M, Netzer M, Dreiseitl S, Baumgartner C. Applied Data Mining: From Biomarker Discovery to Decision Support Systems. In: Trajanoski Z, editor. Computational Medicine. Vienna: Springer Vienna; 2012. pp. 173–184. https://doi.org/10.1007/978-3-7091-0947-2_10

60. He K, Zhang X, Ren S, Sun J. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. 2015 IEEE International Conference on Computer Vision (ICCV). 2015. pp. 1026–1034. https://doi.org/10.1109/ICCV.2015.123

61. Berthold MR, Cebron N, Dill F, Gabriel TR, Kötter T, Meinl T, et al. KNIME: The Konstanz Information Miner. In: Preisach C, Burkhardt H, Schmidt-Thieme L, Decker R, editors. Data Analysis, Machine Learning and Applications. Berlin, Heidelberg: Springer Berlin Heidelberg; 2008. pp. 319–326. https://doi.org/10.1007/978-3-540-78246-9_38

62. Goh GB, Hodas NO, Vishnu A. Deep learning for computational chemistry. Journal of Computational Chemistry. 2017; 38: 1291–1307. https://doi.org/10.1002/jcc.24764 PMID: 28272810

63. Mansouri K, Judson RS. In Silico Study of In Vitro GPCR Assays by QSAR Modeling. In: Benfenati E, editor. In Silico Methods for Predicting Drug Toxicity. New York, NY: Springer; 2016. pp. 361–381. https://doi.org/10.1007/978-1-4939-3609-0_16