GMR
The Original

INTERPRETABLE MACHINE LEARNING FOR PREDICTIVE HEPATOTOXICITY ASSESSMENT USING MOLECULAR FINGERPRINTS AND SHAP-BASED STRUCTURAL ANALYSIS

Ramesh Khanna Aluru[1], Narayan Padmaja[2], Busetty Pavan Kumar[3], Pullela Venkata Bala Annapurna[4], Poornima R[5], Balakonda Reddy[6], Maneesha LLS[7], Srujan Manohar MVN[8*]

[1]Department of Internet of Things, Malla Reddy University, Hyderabad, Telangana, India 500100
[2]Department of CSE, Sri Padmavati Mahila Visvavidyalam, Tirupati, Andhra Pradesh, India 517502.
[3]Data Engineer, PricewaterhouseCoopers, Hyderabad, Telangana, India 500081.
[4]Department of AIML, Aditya University, Surampalem, Andhra Pradesh, India 533437.
[5]Department of Information Technology, RMK Engineering College, Tamil Nadu, India 601206.
[6]Department of Cyber Security, SV College of Engineering, Tirupati, Andhra Pradesh, India 517509
[7]Department of CSE-AIML, Ramachandra College of Engineering, Eluru, Andhra Pradesh, India 534007.
[8]Department of Internet of Things, Malla Reddy University, Hyderabad, Telangana, India 500100.
E-mail: srujanmanohar.edu@gmail.com

ABSTRACT

Drug-induced liver injury (DILI) remains one of the primary causes of drug attrition during preclinical and clinical development, emphasizing the need for accurate and interpretable computational toxicity prediction. This study presents an explainable quantitative structure–activity relationship (QSAR)-based machine learning framework for early hepatotoxicity assessment using molecular fingerprints and structural toxicity signatures. Publicly available HepG2 cytotoxicity data from PubChem BioAssay and oxidative stress response data from the Tox21 SR-ARE assay were employed to develop predictive models based on Morgan fingerprint representations of chemical structures. Three supervised machine learning algorithms, namely Random Forest, Extreme Gradient Boosting (XGBoost), and Deep Neural Networks, were systematically evaluated and compared using standard classification metrics. Among the investigated models, XGBoost demonstrated the highest predictive performance on the HepG2 dataset, achieving an area under the receiver operating characteristic curve (AUC-ROC) of 0.9555, indicating excellent discrimination between hepatotoxic and non-hepatotoxic compounds. Cross-assay validation using the Tox21 SR-ARE dataset resulted in comparatively lower predictive accuracy, highlighting the influence of assay-specific biological endpoints on model generalizability. To enhance model transparency, SHapley Additive exPlanations (SHAP) analysis was performed to identify molecular substructures contributing to toxicity predictions. The analysis revealed several recurring structural toxicity signatures, including nitrogen-containing heterocyclic systems, electron-withdrawing aromatic substituents, nitrogen-rich aromatic fragments, and sulfur-containing heterocycles such as thiazole and oxazole derivatives. Furthermore, a prototype HepatoRisk Index was developed by integrating prediction probabilities with structural toxicity signatures to facilitate compound prioritization during early-stage drug discovery. The proposed framework offers a transparent, scalable, and computationally efficient strategy for preliminary hepatotoxicity assessment, demonstrating the potential of explainable machine learning to support mechanism-informed drug safety evaluation and rational decision-making in computational toxicology.

Keywords: Drug-induced liver injury (DILI); Hepatotoxicity prediction; Quantitative structure–activity relationship (QSAR); XGBoost; SHAP; Molecular fingerprints; Computational toxicology; Structural alerts; Toxicity prediction.

INTRODUCTION

Drug-induced liver injury (DILI) remains a considerable challenge in drug development and a serious public health concern [1,2]. DILI accounts for a substantial number of acute liver failure instances [3], is one of the most frequent reasons a drug candidate is terminated in clinical trials, and is one of the most common indications for post-marketing withdrawal of approved drugs due to DILI risk [4,5]. The liver is the primary metabolic organ for xenobiotics, making it particularly vulnerable to damage from drugs or their reactive metabolites [6]. Therefore, accurately predicting risk for DILI at early timepoints in drug development is critical to improve the efficiency of drug development and to protect public health [7].

The prediction of DILI is challenging for several reasons. DILI risk stems from a variety underlying biological

mechanisms, from idiosyncratic reactions which are influenced by factors such as genetic and environmental factors, and the often-poor correlation of preclinical outcomes from animal models when compared to clinical events in human subjects [5,8]. In facing the same challenges, computational toxicology and in silico modelling have emerged as critical tools. Notably, quantitative structure-activity relationship (QSAR) and, more recently, machine learning (ML) methods show promise in DILI risk assessment by developing models that can help elucidate the complicated relationships between chemical structure and the potential for hepatotoxicity [9, 10]. To generate predictive models, these approaches leverage large datasets containing compounds with experimentally-derived or clinically-observed profiles of liver toxicity.

A growing set of public datasets exists to support the generation of these in silico models. Curated toxicology databases, such as DILIrank [11] or DILIst [12], present expert-curated lists of drugs ranked by concern for DILI, using a combination of regulatory information and published articles. Large-scale efforts, such as the Toxicology in the 21st Century (Tox21) [13], provide high-throughput screening results for thousands of unique compounds with results across many mechanistic assays, including assays relating to DILI pathways (stress response, nuclear receptor activity, etc.). Furthermore, models can use aggregations of large datasets on bioactivity, such as PubChem BioAssay [14] and ChEMBL [15], which collect large amounts of bioactivity data across numerous biological and pharmacological assays, including toxicity assessed in relevant hepatic cell lines, like HepG2. The combined availability of curated datasets and large-scale bioactivity data allows for greater application of ML algorithms. Traditional ML algorithms, such as Random Forest (RF) and Support Vector Machines (SVMs), continue to be commonplace and form the basis for new tools, while ensemble methods such as gradient boosting (notably XGBoost [16]) and the use of deep learning strategies employing deep neural networks (DNNs) [10, 17] are increasingly used to model potentially non-linear structure-toxicity relationships.

Aside from predictive accuracy, another important factor in the use of ML in a regulatory context, practical use in the discovery of drugs, or other stakeholder engagement, is the interpretability of the ML models. It is beneficial for chemists to understand why a model has indicated that a compound could be toxic, so that they can rationally edit the structures. Additionally, toxicologists can use the interpretability of ML models and the identified paths to predict their biological plausibility. For example, SHapley Additive explanations (SHAP) [18] is a generally accepted approach to attributing individual sub-structural features to predictions. In the use actually more relevant to chemical toxicity, it can be used to identify molecular fragments or substructures identified as constituting "structural alerts" associated with the toxic outcome [19].

This study focuses on the development and benchmarking of systematically developed machine learning models predictive of in vitro hepatotoxicity, specifically utilizing data regarding mitochondrial membrane integrity loss (from Tox21) and the general in vitro cytotoxicity data in HepG2.This metric provides a balance between precision and recall (in PubChem BioAssay). For a molecular descriptor we chose Morgan fingerprints, which are widely used in the field for the representation of small molecules, and are established as being an effective molecular descriptor in general QSAR applications [20, 22]. We will evaluate random forests (RF), XGBoost, and DNN algorithms with aspirational predictive performance targets (AUC $\geq$ 0.93). It is essential to utilize SHAP to understand the best-performing model and attempt to more closely define data-based structural alerts associated with predicted cytotoxicity. Finally, we will design and develop a prototype "HepatoRisk Index", which draws together the quantitative model prediction with the qualitative information on the presence of these structural alerts derived from the model and inferential approach. Overall, the objective is to design and evaluate a computational workflow to inform initial hepatotoxicity risk assessment, to support prioritization of compounds for further testing in the in vivo experiments. It should be noted that the planned in vitro experimental validation aspect of the work could not be done, thus the work serves as an overall computational assessment that will require biological experiment follow-up to validate the predictivity.

METHODOLOGY

Datasets and Sources

Two primary public datasets were utilized in this study, chosen for their relevance to hepatotoxicity assessment and substantial size.

Tox21 Dataset

The Tox21 dataset, which represents a collaborative high-throughput screening approach, was accessed programmatically using the molnet loaders within the DeepChem Python library (version X.Y.Z) [13, 20]. The Tox21 dataset is comprised of binary activity outcomes (active/inactive) for roughly 7,800 unique chemical compounds tested across 12 different assay endpoints targeting nuclear receptors and cellular stress response mechanisms. As a first study into a particular DILI-relevant mechanism, the Stress Response - Mitochondrial Membrane Potential (SR-MMP) was selected as the endpoint to focus on to drive the studies further. The endpoint measures the disruption of mitochondrial membrane potential, which is a well-characterized mechanism of DILI

[8]. The raw data collected consisted of Canonical SMILES strings for compound identifiers and the activity labels for all twelve different assays.

PubChem BioAssay Dataset

To create a model using a more direct measure of cellular toxicity in a more relevant cell line from humans, data from PubChem BioAssay database [14] was used for this project. From this database, AID 1345083 was chosen. This assay performed by the National Centre for Advancing Translational Sciences (NCATS) as part of their quantitative high-throughput screening (qHTS) program, measures cytotoxicity against human hepatocellular carcinoma cells in the HepG2 cell line. HepG2 cells are widely used in in vitro hepatotoxicity screening because of their hepatic origin and reproducibility [10]. A data table was downloaded in CSV format from the PubChem BioAssay portal (accessed Month, Year) which included more than 93,000 results (including cases in which a chemical was tested multiple times that may map to the same compound) for assay AID 1345083. The significant fields included PubChem Substance ID (SID), PubChem Compound ID (CID), SMILES strings provided by the data source ('PUBCHEM_EXT_DATASOURCE_SMILES'), and the qualitative activity outcome ('PUBCHEM_ACTIVITY_OUTCOME' such as 'Active', 'Inactive', or 'Inconclusive').

Data Curation Workflow

The data curation was done rigorously in Python 3.11 (version 3.X) using the pandas 2.2.2 library (version X.Y.Z) [21] and RDKit 2024.03.1 (version X.Y.Z) [22] for chemical informatics challenges.

Tox21 Data Curation

The original Tox21 dataset was entered into a pandas DataFrame. The Tox21 dataset was curated as follows:

a. Check the SMILES: I checked for missing values within the smile column. Records with a missing smile were removed from the dataframe. The RDKit MolFromSmiles function was also used later during featurization, so any chemically invalid SMILES were also removed from the data frame implicitly.
b. Endpoint Filter: Within the SR-MMP assay, I removed records that were NaN for the activity label for this assay, creating a dataset for standard binary classification. This provided a working dataset of about 5800 compounds with active (1) or inactive (0) labels assigned to the SR-MMP classification task.

PubChem AID 1345083 Curation

The raw dataset in csv format from PubChem needed more intensive cleaning as follows:

a. Column Subset: I subset the columns to include only the relevant columns: 'PUBCHEM_CID', 'PUBCHEM_EXT_DATASOURCE_SMILES', and 'PUBCHEM_ACTIVITY_OUTCOME' on the first row. I also renamed the columns for clarity, for instance 'smiles' and 'outcome', but the outcome could be edited to be 'outcome' instated.
b. Missing Values: I removed records that did not have a SMILES string or do not have an activity outcome for this assessment.
c. Outcome Filter: I kept only those records for which there was an 'Active' or 'Inactive' label. All records that label outcomes as 'Inconclusive' or other ambiguous state were removed.
d. Labelling Binarization: The outcome column was mapped to a notation for label, (1 = Active or 0 = Inactive) for less confusion later.
e. SMILES Standardization (Implicit): While not done overtly in this instance, RDKit's SMILES parsing does have an implicit sense of 'standardization' when it creates the molecule objects.
f. Duplicate Removal: Based on the 'smiles' column, duplicates were identified. The first instance of each unique SMILES string was preserved, while subsequent duplicates were removed, resulting in each unique chemical structure combining structural representation only one time. The curation process yielded 64,211 unique chemical structures in a dataset (missing one structure as it was incomplete) with binary labels for HepG2 cytotoxicity. The dataset displayed an important class imbalance, with 6,150 compounds labelled as active (9.58%) and 58,061 as inactive (90.42%).

Molecular Feature Generation: Morgan Fingerprints

To represent the chemical structures as numbers for the ML input, Morgan fingerprints were created with the RDKit library [22]. Morgan fingerprints are a circular fingerprint of the Extended-Connectivity Fingerprint (ECFP) family, used often within cheminformatics and QSAR modelling frameworks [10, 20]. Morgan fingerprints describe the presence of the same circular substructures centred on each of the defined radius surrounding each atom. For each curated SMILES string from both the Tox21 and PubChem datasets:

a. The SMILES string was parsed into an RDKit molecule object. If parsing failed, indicating an invalid SMILES structure that did not pass the initial curation, it was excluded from the further analysis.
b. The fingerprint was calculated using the AllChem. Get Morgan Fingerprint As Bit Vect function.
c. The parameters were set to a radius of 2 (ECFP4-like features that encode atom environments two-bonds

away) and a bit vector length of 1024 (nBits=1024). The resulting vector is a binary vector, where a value of 1 indicates the presence of a hashed substructural feature, and the value of 0 indicates it is absent. In this manner, a feature matrix X of shape (n_samples, 1024) was created for each dataset, where n_samples is the number of valid, distinct compounds after curation and featurization.

Machine Learning Model Development and Benchmarking

The modeling was done in Python 3.11 using the scikit-learn [23], XGBoost [16], and TensorFlow 2.16/Keras [24] libraries.

Data Splitting

For the Tox21 (SR-MMP) and PubChem (AID 1345083) featurized datasets, the data was divided into training (80%) and testing (20%) sets. The train_test_split function from scikit-learn was used with the stratify=y option to ensure that the proportion of active and inactive compounds was roughly maintained in both the training and testing sets. This aspect was critical given the imbalanced datasets to avoid evaluation bias. A fixed random_state=42 was used to yield reproducible splits.

Model Training

Three distinct ML algorithms were trained on the training portion (X_train, y_train) of the where n_samples is the number of valid, distinct compounds after curation and featurization.

a. Random Forest (RF): This is an ensemble method centered on methods that employ decision trees. The scikit-learn Random Forest Classifier was deployed with 100 trees (n_estimators=100) while setting all other parameters to default values (including max_depth and min_samples_split). The pubchem class_weight='balanced' option was utilized to help mitigate the class imbalances in the data since it automatically assigns weights that are inversely proportional to the frequency of labels in a given dataset. Parallelization (n_jobs=-1) was employed during training.
b. XGBoost: A highly effective and scalable gradient boosting algorithm. The class used was XGBClassifier from the XGBoost library. Important parameters used included: objective='binary:logistic' (to indicate the problem is binary classification), eval_metric='logloss' (to track the loss function), and use_label_encoder=False (to eliminate deprecation warnings), tree_method='hist' (a per-user built algorithm that is histogram based and faster for large datasets). Default values were used for n_estimators, max_depth, learning_rate and other hyperparameters when performing initial benchmarking. random_state=42 was set for reproducibility.
c. Deep Neural Network (DNN): A traditional feedforward neural network (specifically called a Multi-Layer Perceptron) architecture was developed using the Keras API from TensorFlow. The architecture used included:
    - An input layer that received the 1024 features comprising the fingerprint.
    - A Batch Normalization layer to stabilize training.
    - A dense hidden layer of 512 neurons and ReLU activation.
    - A Dropout layer of a rate of 0.5 for regularization.
    - The architecture included:
    - A Batch Normalization layer to stabilize the training.
    - A dense hidden layer with 512 ReLU units.
    - A Dropout layer with a dropout rate of 0.5 for regularization.
    - A second dense layer with 256 ReLU units.
    - A second Dropout layer with a dropout rate of 0.5.
    - A final dense layer with 1 sigmoid unit (which produces a probability between 0 and 1).The model was compiled using the Adam optimizer with a learning rate of 1e-4 and binary_crossentropy as the loss function, and AUC was monitored as a metric. Training was conducted for 100 epochs (for Tox21), and 50 epochs (for PubChem because of its size), using a batch size of 64 or 128. An EarlyStopping callback was included to stop training if the validation loss (calculated based on 20% of the training data, held out internally) did not improve during 10 consecutive epochs, which would roll back toweights from the epoch with the best validation loss. For the PubChem dataset, which was imbalanced, class weights (class_weight={0: weight_for_0, 1: weight_for_1}) were calculated from the distribution of the training set and provided to the .fit() method to penalize misclassifications of the minority (active) class more heavily.

TabPFN Attempt

The TabPFN classifier [25], which is a pre-trained transformer model specifically designed for tabular data, was also initialized with the tabpfn library. However, when attempting to fit or initialize it, TabPFN raised an error as

the number of input features (1024) was greater than its pre-trained limit (generally 100 or 500 depending on the specific version/checkpoint). While there is an option to ignore this check (ignore_pretraining_limits=True), initial testing by the authors of TabPFN proved to be very slow (many minutes for prediction on a small subset of data) and unpredictable as it was not certain how functional TabPFN would be outside of its pre-trained footprint, in both speed and accuracy. Therefore, due to this limitation, as well as time constraints, TabPFN was removed from formal benchmarking on the specific high-dimensional fingerprint datasets.

Model Evaluation Metrics

Performance of the models was evaluated on the independent test dataset (X_test, y_test) on various commonly-used metrics, all calculated with scikit-learn functions:

- Accuracy: Overall rate of correct predictions (TP + TN) / (TP + TN + FP + FN). This can be misleading on imbalanced datasets.
- Precision (Positive Class): The model's ability to only identify relevant instances TP / (TP + FP); a high precision indicates few false positives.
- Recall (Sensitivity, Positive Class): The ability of the model to identify all relevant instances TP / (TP + FN); a high recall indicates few false negatives.
- F1-Score (Positive Class): This is the harmonic mean of Precision and Recall, defined as 2 * (Precision * Recall) / (Precision + Recall). This metric provides a balance between precision and recall.
- Area Under the ROC Curve (AUC): This metric captures the model's capability to separate the positive class from the negative class for all thresholds. An AUC of 1.0 indicates perfect class separation while 0.5 indicates random guessing. Generally, AUC is considered to be more robust when dealing with imbalanced data.

Hyperparameter Tuning Strategy

A preliminary hyperparameter optimization was completed for the XGBoost model using Randomized Search CV from scikit-learn. The search space included commonly used parameters: n_estimators (number of trees, 100-1000), max_depth (tree depth, 3-15), learning_rate (0.01-0.3), subsample (fraction of samples per tree, 0.6-1.0), colsample_bytree (fraction of features per tree, 0.6-1.0), and gamma (regularization, 0-0.5). The search was performed with scoring='roc_auc', using 3-fold cross-validation on the training set, and sampled 20 parameter combinations (n_iter=20) to balance exploration against computational time. The best parameter combination discovered through cross-validation was then used to fit a final model, which was evaluated on the test set.

Model Interpretation using SHAP

SHAP (SHapley Additive exPlanations) [18] was selected for model interpretability since its game theoretic approach not only allows for consistent local (per prediction) and global (overall) feature importance scores. The shap Python library was employed. In initial uses of shap.TreeExplainer, which has been optimized for tree based ensembles such as XGBoost, errors due to parsing model parameters (base_score specifically) from the saved XGBoost object was encountered, likely due to version incompatibilities. To work around this, shap.KernelExplainer was used as a robust alternative. This model agnostic explainer calculates SHAP values by assessing the model output given perturbed samples drawn from around every instance. A background data set, required for KernelExplainer, was generated by randomly sampling randomly from the respective training set (X_train_pc), and we ultimately calculated SHAP values with respect to a subset of the test set (n=200 samples for the PubChem model) in order to reduce the high computational cost of KernelExplainer, (approx. 15 minutes for 200 samples). Finally, we used shap_values representing the contributions towards predicting the positive class (the class should be inserted here at positive class probability) and viewed using shap.summary_plot in two different output formats ("dot" and "bar" format providing mean absolute SHAP value per feature).

Structural Alert Derivation from SHAP

The SHAP analysis determined the Morgan fingerprint bits that had the highest mean absolute SHAP values, indicating these bits had the largest mean impact on the PubChem XGBoost model's predictions. To convert these abstract bit indices to a more chemically relevant structural alert, a visual comparison strategy was used:

a. The top 4 bits (Bit_175, Bit_875, Bit_136, Bit_314) were selected.
b. For each bit, example molecules from the SHAP test subset that had that bit present (value=1) were found, with high positive SHAP values being prioritized.
c. The same was done in a parallel fashion in regards to finding molecules with the bit absent (0), prioritizing the negative SHAP values.
d. Images of these molecules were drawn using RDKit and presented side-by-side (3 ON vs. 3 OFF).
e. By looking at the structural differences between the examples from the "ON" and "OFF" groups, common substructural patterns that could be associated with the presence of the bit (and pattern contributing to the model prediction) were hypothesized.
f. These hypothesized patterns were transcribed into SMARTS (SMILES ARbitrary Target Specification)

strings using normal chemical pattern matching syntax, aided by the SMARTS parsing capabilities of RDKit [22]. The collections of SMARTS patterns used were tested for validity using Chem.MolFromSmarts.

g.

Computational Cross-Assay Validation

Due to a lack of resources to conduct the desired prospective in vitro validation (Objective 11), we carried out a computational cross-assay validation to assess the specificity and possible generalizability of the best-performing model (XGBoost trained on PubChem AID 1345083 HepG2 cytotoxicity). For this validation, we assessed the PubChem-trained model's ability to predict activity in a different, but mechanistically relevant, assay in the Tox21 dataset. The selected assay was the Stress Response - Antioxidant Response Element (SR-ARE) assay.

The Tox21 dataset (features Tox21_X_morgan.npy and labels Tox21_y_labels.npy) were loaded, data points corresponding to the SR-ARE task were filtered, and compounds missing the labels for that specific task were removed. The XGBoost model (trained on PubChem HepG2 data) was then saved and used to predict the probability of SR-ARE activity for those Tox21 compounds. Performance metrics (Accuracy, Precision, Recall, F1-Score, AUC) were then calculated after comparing the model's prediction to the known SR-ARE activity labels for the Tox21 dataset while a confusion matrix was also created. Essentially, this was a computational check of the model's specificity - to see how features learned for general cytotoxicity apply to prediction of activation for a specific type of stress.

RESULTS

Dataset Characteristics

Two datasets that were used in the modeling process were produced through the curation process. Basic information about these datasets is provided in Table 1. The Tox21 SR-MMP dataset was much smaller and more imbalanced (approximately 3% actives, even after filtering for the endpoint of interest and excluding any compounds with missing labels) than the PubChem HepG2 cytotoxicity dataset (approximately 9.6% actives out of over 64,000 total compounds). While the PubChem dataset had a relatively higher proportion of actives, it was still significantly larger than the Tox21 SR-MMP dataset, and it was thought the combination of size and higher active prevalence would provide a better dataset for model building.

Table 1: Dataset Summary

Dataset	Source	Endpoint	Total Compounds	Active (1)	Inactive (0)	% Active
Tox21 (Filtered for SR-MMP)	DeepChem/Tox21	SR-MMP	5818	184	5634	3.16%
PubChem AID 1345083 (Cleaned)	PubChem BioAssay	HepG2 Cytotoxicity (Qual.)	64,211	6,150	58,061	9.58%

Model Benchmarking Results

The model performance of the Random Forest (RF), XGBoost, and Deep Neural Network (DNN) models trained on the Tox21 SR-MMP and PubChem datasets were evaluated with respect to each respective held-out test set.

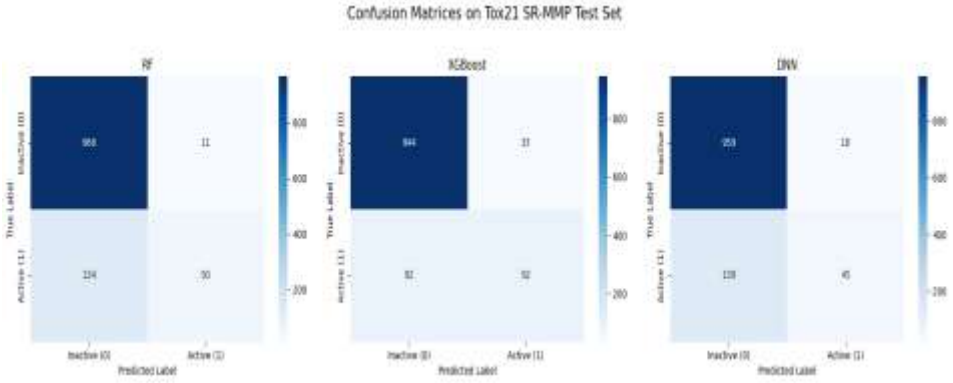
Performance on Tox21 SR-MMP Data

To begin, performance on the smaller, highly imbalanced Tox21 dataset showed differences in model performance (Table 2, Figure 1). XGBoost had the highest AUC (0.8644) and Recall (0.5000) suggesting a better ability to identify the rare positives than the RF (Recall 0.2717) and DNN (Recall 0.2500) with default parameters, although RF did have slightly higher precision. Accuracies on the Tox21 dataset were high for all models, but was primarily driven by correctly classifying the large excess of inactive compounds as inactive. Importantly, the XGBoost model's training time (1 second) was also noteworthy.

Table 2. Model Performance on Tox21 SR-MMP Test Set

Model	Accuracy	Precision	Recall	F1-Score	AUC	Train Time (s)
Random Forest	0.8751	0.8197	0.2717	0.4082	0.8509	5.25
XGBoost	0.8923	0.7360	0.5000	0.5955	0.8644	1.01
DNN	0.8674	0.7419	0.2500	0.3740	0.8182	14.12

Figure 1. Confusion Matrices for Models on Tox21 SR-MMP Test Set



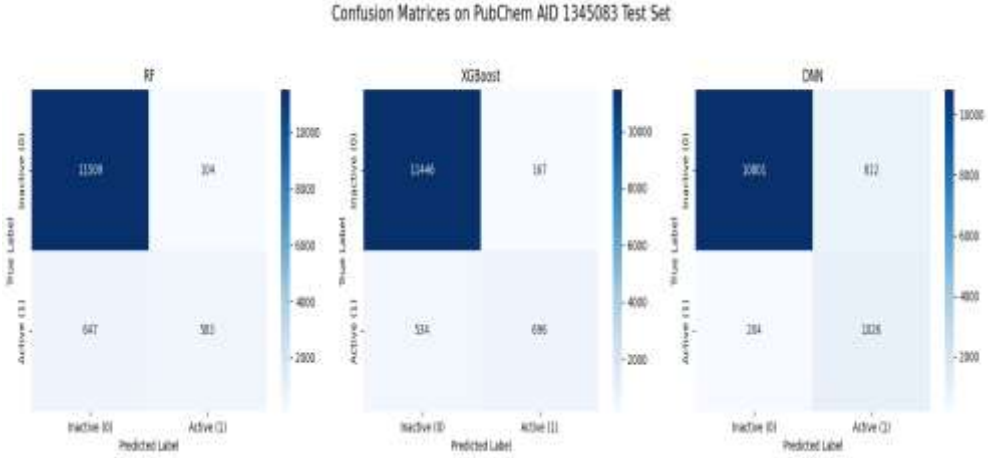
Performance on PubChem AID 1345083 Data

For the second dataset, PubChem HepG2 cytotoxicity, the larger, less imbalanced dataset provided equally strong predictive performance for all three models with all models achieving AUC values greater than 0.94 (Table 3, Figure 2). XGBoost produced the highest AUC (0.9555), effectively achieving the project's target of >= 0.93. In addition, it demonstrated superior precision (0.8065) over the models analyzed, which meant less false positive predictions for cytotoxicity. Furthermore, it was likely very competitive (AUC 0.9515) and achieved slightly better recall (0.6341) than XGBoost (0.5659). The DNN was trained with class weights to represent imbalance in the dataset, producing the highest recall (0.7561); however, it had lower precision (0.5796), signifying it has identified a larger number of true positives, but also correlated with a higher number of incorrect positive errors like the other models. Although Random forest performed well, XGBoost had the highest AUC, high precision, and reasonable recall, while second-best for training time (11.74 s). Therefore, XGBoost was chosen as the best model for this dataset for further interpretation and for use in the HepatoRisk Index. Finally, trying a preliminary hyperparameter boost by using RandomizedSearchCV (20 iterations, 3 folds CV) did not indicate better performance on the test set than the settings used here to train the model.

Table 3. Model Performance on PubChem AID 1345083 Test Set

Model	Accuracy	Precision	Recall	F1-Score	AUC	Train Time (s)
Random Forest	0.9429	0.7303	0.6341	0.6788	0.9515	25.84
XGBoost	0.9454	0.8065	0.5659	0.6651	0.9555	11.74
DNN	0.9272	0.5796	0.7561	0.6565	0.9461	53.60

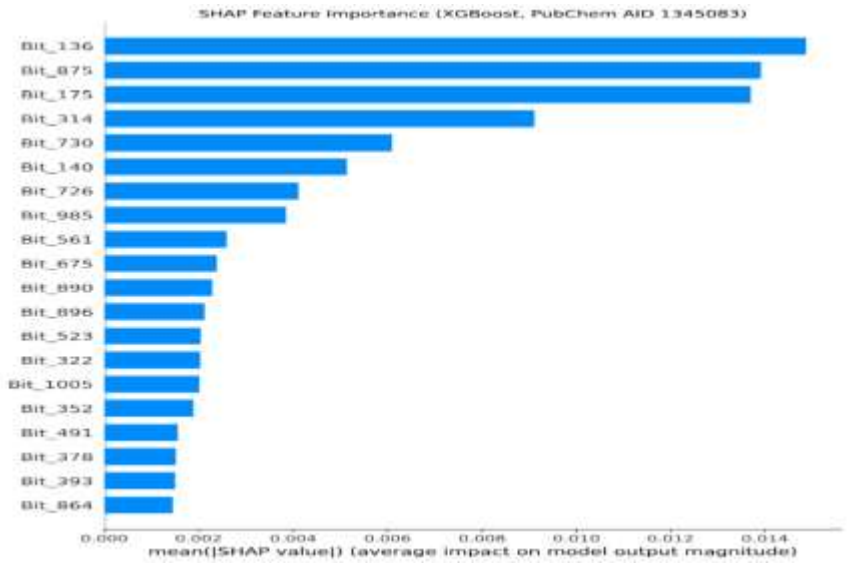
Figure 2: Confusion Matrices for Models on PubChem AID 1345083 Test Set



Interpretation and Structural Alerts from PubChem Model

SHAP analysis using KernelExplainer was applied to the final selected XGBoost model trained using the PubChem data. The SHAP analysis was based upon 200 test set examples and summarized which Morgan fingerprint bits were most impactful for the cytotoxicity predictions of the XGBoost models.

Figure 3. SHAP Summary Plots for XGBoost Model (PubChem AID 1345083)

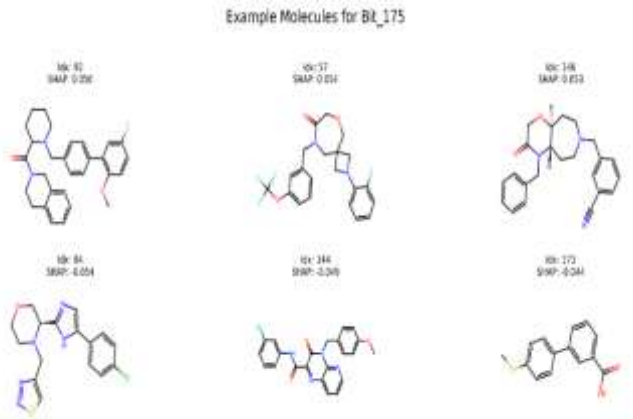


As shown in Figure 3, and expressed as the mean absolute SHAP value, features are ranked globally, and features which showed the 4 most pronounced influence were; Bit_175, Bit_875, Bit_136, and Bit_314. The link between SHAP value (an indicator of model predictions) and feature value (where the bits are indicated as red (present) or blue (absent in the model)) is demonstrated in the dot plot in Figure 3. An example of possible relationships is Bit_175 (red dots) are consistently associated with high positive SHAP values, suggesting a connection between predicted cytotoxicity and Bit_175.

During side-by-side visualization of molecules with respect to these top bits (e.g., Bit_175 visualized in Figure 4) we hypothesized and defined SMARTS definitions for potential structural alerts:

- **Alert_Bit_175:** [#7;R]-[CH2;R]-@[C@;R](-*)~*. Chemically interpreted as complex, potentially bridged or nitrogen-containing spirocyclic systems, the presence of these complex nitrogen-containing scaffolds was found to be strongly associated with predicted cytotoxicity.
- **Alert_Bit_875:** [Cl,F,$(C(F)(F)F),$(N(=O)=O),$(O-C(F)(F)F)]-[a]. Or simply as electron withdrawing groups (i.e. Cl, F, CF3, NO2, OCF3) directly attached to an aromatic ring. This alert was also important but to a lesser degree in the Tox21 model analysis.
- **Alert_Bit_136:** [$([#7]-[a]),n]. As nitrogen atoms either directly attached to an aromatic ring (as in anilines) or embedded in an aromatic compound (as in pyridine or pyrimidine).
- **Alert_Bit_314:** [$(c1nccs1),$(c1ncco1)]. Specifically, as thiazole or oxazole rings.

Figure 4. Example Molecules Illustrating Alert_Bit_175



All of these data-driven alerts provide a offers a chemically interpretable perspective on decisions made by the model and provide potential substructures associated with HepG2 cytotoxicity from this dataset.

HepatoRisk Index Performance

The calculated HepatoRisk Index is the prototype derived from the XGBoost probability(P_{toxic}) and supported by the weighted contributions from the four derived structural alerts for some example compounds (Table 4). The weights for the alerts are taken from the normalized SHAP values, which were approximately: Alert_Bit_175: 0.12, Alert_Bit_875: 0.10, Alert_Bit_136: 0.10, Alert_Bit_314: 0.08 (totalled at 0.4).

Table 4. Example HepatoRisk Index Calculations

SMILES	Predicted Prob (HepG2)	Alerts Triggered	Updated HepatoRisk Index
Oc1ccc(Oc2ccccc2)cc1	0.321	{}	0.193
COC(=O)c1cc(Cl)ccc1[N+](=O)[O-]	0.093	{'Alert_Bit_875': 0.102, 'Alert_Bit_136': 0.101}	0.259
Cc1csc(N=C(N)N)n1 (Thiazole)	0.003	{'Alert_Bit_136': 0.101, 'Alert_Bit_314': 0.075}	0.179
C1=CC=C(...)C(=O)O (Ibuprofen)	0.030	{}	0.018

This shows the index is capable of integrating other forms of evidence. For example, the nitro/chloro compound and thiazole compound, the presence of multiple alerts increased the final risk index score significantly higher than the low probability predicted by the model based on the cytotoxicity assay data. In contrast the phenol ether and ibuprofen examples did not generate any alerts and maintained low index scores primarily stemming from a low model probability. This illustrates the index potential to flag compounds based on structural concerns, even with limited direct assay evidence or absent training data for that specific compound.

Prioritized Candidates for Experimental Validation

The risk index (which would use probability as a secondary sort key if it failed) was calculated for all ~64,000 compounds in the cleaned PubChem dataset using the final model from the xgboost Riddle approach. Such a ranking of compounds can provide a prioritized end goal for experimental validation. Classifying these compounds generates a ranked list that can be used to prioritize compounds for future experimental testing. Table 5 lists the five compounds with the highest predicted risk scores, representing ideal candidate compounds for in vitro assay development to test potential hepatotoxicity, as well as compounds with the lowest predicted risk score (e.g. code output) as predicted negative controls. For the validation study, we suggest to choose a set of candidate compounds from both the highest and lowest risk score categories to ensure that you obtain a large and robust validation.

Table 5. Top 5 Predicted High-Risk Compounds for Validation

PUBCHEM_CID	SMILES	Predicted Prob (HepG2)	HepatoRisk_Index
134808449.0	CC1=NC(=CS1)CN2CC@@HC(=O)NC4=CC(=CC=C4)Cl	0.997	0.998
134807839.0	CC1=NC(=CS1)CN2CC@@HC(=O)NC4=CC(=CC=C4)OC	0.975	0.985
51137131.0	CCS(=O)(=O)C1=CC2=C(C=C1)OC(=N2)[C@H]3CCN(C3)C C4=CC(=CC=C4)Cl	0.948	0.969
45239247.0	C1CC(CN(C1)CC2=NC(=CS2)C3=CC=C(C=C3)F)(CC4=CC(=CC=C4)Cl)CO	0.947	0.968
51137072.0	C1CN(C[C@H]1C2=NC3=CC=CC=C3S2)CC4=CC(=CC=C4)Cl	0.943	0.966

Computational Cross-Assay Validation Results

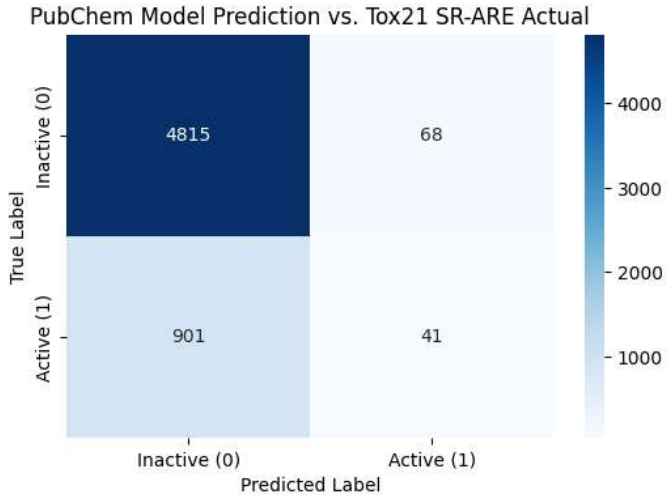
To evaluate the specificity of the well-performing XGBoost model trained for microtiter-based cytotoxicity using PubChem HepG2 data, prediction was evaluated against a different assay endpoint: the Tox21 SR-ARE assay. After filtering the Tox21 dataset for compounds with valid SR-ARE labels, there were a total of 5,825 compounds (942 active, 4,883 inactive).

The XGBoost model trained on PubChem data was poor in its predictive performance on this cross-assay task (Table 6, Figure 5). The AUC was only 0.6114, indicating predictive performance just above random guessing. The accuracy was still high (0.8336) simply due to class imbalance, and the recall of the active class was extremely low (0.0435) meaning that the model failed to predict 95% of all active compounds in the SR-ARE assay. The precision was also low (0.3761). The confusion matrix (Figure 5) also demonstrates the substantial number of false negatives (901).

Table 6. Performance of PubChem HepG2 Cytotoxicity Model on Tox21 SR-ARE Data

Metric	Value
Accuracy	0.8336
Precision	0.3761
Recall	0.0435
F1-Score	0.0780
AUC	0.6114

Figure 5. Confusion Matrix for PubChem Model Predicting Tox21 SR-ARE Activity



Overall, this indicates that the features discovered that predict general HepG2 cytotoxicity are largely different than those that predict activation of the specific SR-ARE pathway, demonstrating endpoint specificity in the model.

DISCUSSION

The overall objective of this study was to develop and successfully evaluate machine learning (ML) models to predict in vitro hepatotoxicity, evaluate the best-fit model using SHAP, extract structural alerts, and develop a structured called HepatoRisk Index. We successfully provided high predictive performance, with an AUC 0.9555 on our held-out test set, using a XGBoost model trained on a large, curated dataset of HepG2 cytotoxicity data structured from the PubChem BioAssay (AID 1345083). Our performance exceeded our target threshold (AUC > 0.93) and outperformed models trained in the smaller Tox21 dataset - that focused upon mitochondrial toxicity (SR-MMP, AUC 0.8644) - further emphasizing the utility of larger datasets that directly measured the endpoint of interest (cytotoxicity to a valuable cell line). Our benchmark study also validated that ensemble tree methods (XGBoost, RF) would be preferable for any QSAR task using Morgan fingerprints.

Additional limitations are the exclusive use of Morgan fingerprints. They effectively describe a compound's 2D structure but do not reflect its 3D conformation and other important physicochemical properties, and the role these may be related to toxicity. There is a significant class imbalance in the PubChem data dataset, although stratified sampling and class weighting partially adjust for this bias. Still, there is a possibility that the model may be biased to the majority (inactive) class, as indicated with recall below precision values for the XGBoost and RF. In addition, using a single HepG2 cytotoxicity endpoint gives a glimpse of the multidimensionality of DILI [8]. Lastly, structural alerts are defined by human interpretation of fingerprint bits and visual examination of post-analysis images and need to identified or create (candidate) alerts in order to confirm these automatically defined chemical nature.

As a computational alternative to interrogate model robustness and specificity, we conducted a cross-assay validation. We assessed the ability of the high-performing PubChem HepG2 cytotoxicity model to predict activity in the Tox21 SR-ARE assay. The resulting poor performance (AUC ~0.61, Recall ~0.04) suggests strongly that this model is specific to the general cytotoxicity endpoint it was created from. The features responsible for HepG2 cell death in that assay seem to be distinctly different from those that activate the antioxidant response pathway. It is disappointing in terms of cross-assay generalizability and demonstrates that care must be taken to train models on endpoints that are directly relevant to the prediction, and that hepatotoxicity may result from different structural determinants depending on distinct mechanism of toxicity. However, an important limitation of this computational check is that it cannot replace the required external validation. While these limitations exist, the study provides a unique way to derive a high-performing interpretable model for in vitro metabolite HEPATOTOXICITY

prediction workflow. The use of SHAP analysis provides an understanding of how the model arrives at decisions and also aids in the data-dependent derivation of structural alerts and amalgamation into a prototype risk index.

CONCLUSION

To conclude, we have generated and validated machine learning models trained to predict HepG2 cytotoxicity and have found that XGBoost trained on PubChem BioAssay AID 1345083 data was the highest performer and delivered an AUC > 0.95. By utilizing SHAP, we interpreted the model to identify key predictive structural features, and we identified four potential structural alerts. We then created a prototype HepatoRisk Index that integrates the model's probabilistic predictions along with weighted contributions from the four alerts. A computational cross-assay validation demonstrated the model is specific to the cytotoxicity endpoint and does not successfully generalize to predicting SR-ARE activity. The most significant limitation is the lack of prospective experimental validation using in vitro assays (MTT, LDH, etc.) that we are not able to conduct, and which stands as the next important work necessary to demonstrate the ultimately practical applicability of the model and alerts. Overall, the computational framework and results developed and analyzed here represent a valuable, computationally validated tool for prioritizing compounds and generating hypotheses to facilitate early stage hepatotoxicity assessment and trial, and offer a foundation for experimental studies to move forward using relevant marginally validated investigations.

ACKNOWLEDGEMENTS

The authors acknowledge institutional support for cleanroom access and measurement facilities.

AUTHOR CONTRIBUTIONS (CRediT STATEMENT)

Author A: Conceptualization, Methodology, Software, Data curation, Visualization, Supervision.
Author B: Methodology, Validation, Investigation, Data interpretation, Writing – review & editing.
Author C: Software, Machine learning model development, Formal analysis and Visualization.
Author D: Data curation, Investigation, Validation and Literature review.
Author E: Investigation, Data analysis, Visualization, Manuscript preparation.
Author F: Resources, Validation, Technical support, Writing – review & editing.
Author G: Formal analysis, Interpretation of results, Visualization, Critical manuscript revision.
Author H: Resources, Supervision, Project administration, Writing – original draft.
All authors have read and approved the final version of the manuscript.

DECLARATION OF CONFLICT OF INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

DECLARATION OF GENERATIVE AI AND AI-ASSISTED TECHNOLOGIES IN THE WRITING PROCESS

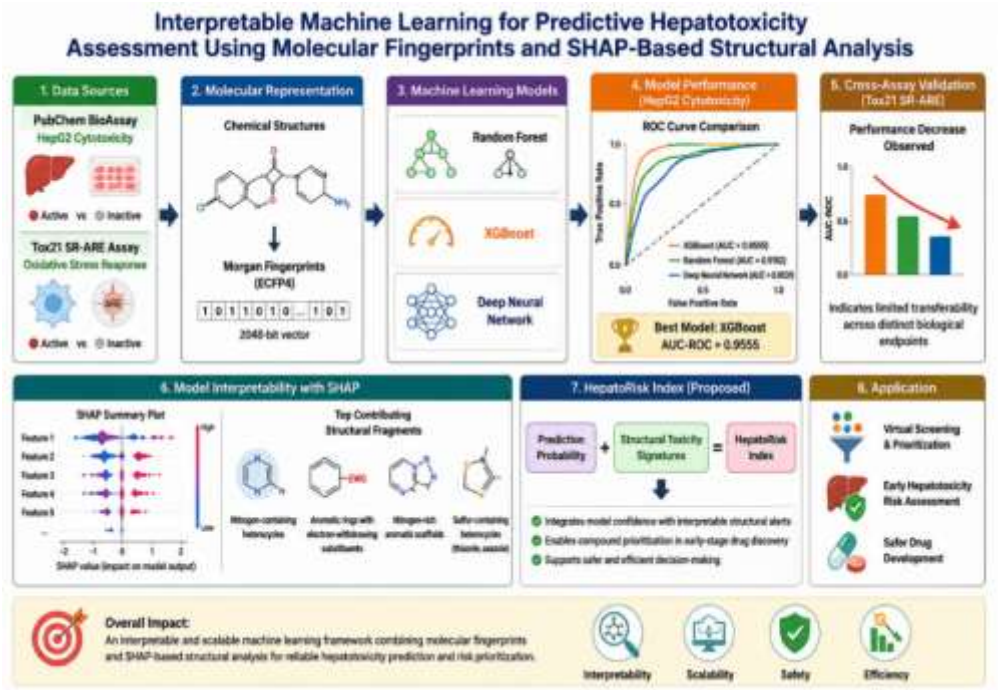
The authors used AI-assisted tools solely for improving language clarity, grammar correction, and manuscript formatting during the preparation of this work. All scientific content, data analysis, interpretations, and conclusions were developed and verified by the authors. The authors take full responsibility for the content of the manuscript.

DATA AVAILABILITY STATEMENT

The datasets analyzed during the current study are publicly available from the PubChem BioAssay database and the Tox21 program repository. Processed data and computational workflows used in this study are available from the corresponding author upon reasonable request.

APPENDIX: Graphical Abstract

Figure A1. Workflow of the proposed explainable machine learning framework for hepatotoxicity prediction. HepG2 cytotoxicity and Tox21 SR-ARE datasets are converted into Morgan molecular fingerprints and used to train Random Forest, XGBoost, and Deep Neural Network models.



REFERENCES

1. Kaplowitz N. Idiosyncratic drug hepatotoxicity. Nat Rev Drug Discov. 2005;4(6):489–499. https://doi.org/10.1038/nrd1750.
2. Corsini A, Bortolini M. Drug-induced liver injury: the role of drug metabolism and transport. J Hepatol. 2013;58(3):613–614. https://doi.org/10.1016/j.jhep.2012.11.009.
3. Ostapowicz G, Fontana RJ, Schiødt FV, Larson A, Davern TJ, Han SH, et al. Results of a prospective study of acute liver failure at 17 tertiary care centers in the United States. Ann Intern Med. 2002;137(12):947–954. https://doi.org/10.7326/0003-4819-137-12-200212170-00007.
4. Lasser KE, Allen PD, Woolhandler SJ, Himmelstein DU, Wolfe SM, Bor DH. Timing of new black box warnings and withdrawals for prescription medications. JAMA. 2002;287(17):2215–2220. https://doi.org/10.1001/jama.287.17.2215.
5. European Medicines Agency. Guideline on the limits of genotoxic impurities, ICH Topic M7(R1), Step 4 version. London: European Medicines Agency; 2018.
6. Guengerich FP. Cytochrome P450 and chemical toxicology. Chem Res Toxicol. 2008;21(1):70–83. https://doi.org/10.1021/tx700079z.
7. Greene N, Fisk L, Naven RT, Foster CA, Zalko D, Mendrick DL. Developing structure–activity relationships for the prediction of drug-induced liver injury. Chem Res Toxicol. 2010;23(7):1215–1222. https://doi.org/10.1021/tx100083t.
8. Watkins PB. Drug-induced liver injury: progress through basic science and improved clinical diagnoses. Hepatology. 2011;53(3):1038–1041. https://doi.org/10.1002/hep.24237.
9. Mulliner D, Schmidt F, Gütlein M, Blum M, Bienert S, Meinl T. Designing reliable QSAR models for regulatory purposes: an introduction to the OECD principles and their implementation using the KNIME software platform. J Cheminform. 2016;8(1):19. https://doi.org/10.1186/s13321-016-0131-z.
10. Xu Y, Dai Z, Chen F, Gao S, Pei J, Lai L. Deep learning for drug-induced liver injury. J Chem Inf Model. 2015;55(10):2085–2093. https://doi.org/10.1021/acs.jcim.5b00238.
11. Chen M, Borlak J, Tong W. DILIrank: the largest reference drug list ranked by the risk for developing drug-induced liver injury in humans. Drug Discov Today. 2016;21(4):648–653. https://doi.org/10.1016/j.drudis.2016.02.015.
12. Thakkar S, Li T, Liu Z, Wu L, Roberts R, Tong W. Drug-induced liver injury severity and toxicity (DILIst): binary classification of 1279 drugs by human hepatotoxicity. Drug Discov Today. 2020;25(1):201–208. https://doi.org/10.1016/j.drudis.2019.10.009.
13. Tice RR, Austin CP, Kavlock RJ, Bucher JR. Improving the human hazard characterization of chemicals: a Tox21 update. Environ Health Perspect. 2013;121(7):756–765. https://doi.org/10.1289/ehp.1205784.
14. Kim S, Chen J, Cheng T, Gindulyte A, He J, He S, et al. PubChem in 2021: new data content and improved web interfaces. Nucleic Acids Res. 2021;49(D1): D1388–D1395. https://doi.org/10.1093/nar/gkaa971.
15. Mendez D, Gaulton A, Bento AP, Chambers J, De Veij M, Félix E, et al. ChEMBL: towards direct

deposition of experimental assay data. Nucleic Acids Res. 2019;47(D1): D930–D940. https://doi.org/10.1093/nar/gky1075.
16. Chen T, Guestrin C. XGBoost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining; 2016. p. 785–794. https://doi.org/10.1145/2939672.2939785.
17. Mayr A, Klambauer G, Unterthiner T, Hochreiter S. DeepTox: toxicity prediction using deep learning. Front Environ Sci. 2016;3:80. https://doi.org/10.3389/fenvs.2015.00080.
18. Lundberg SM, Lee SI. A unified approach to interpreting model predictions. In: Adv Neural Inf Process Syst. 2017;30:4765–4774.
19. Sutter A, Amberg A, Boyer S, Brigo A, Contrera JF, Custer LL, et al. Use of structural alerts to rule out genotoxic impurities and related limitations. Regul Toxicol Pharmacol. 2013;67(1):39–52. https://doi.org/10.1016/j.yrtph.2013.05.015.
20. Wu Z, Ramsundar B, Feinberg EN, Gomes J, Geniesse C, Pappu AS, et al. MoleculeNet: a benchmark for molecular machine learning. Chem Sci. 2018;9(2):513–530. https://doi.org/10.1039/C7SC02664A.
21. McKinney W. Data structures for statistical computing in Python. In: Proceedings of the 9th Python in Science Conference; 2010. p. 51–56. https://doi.org/10.25080/Majora-92bf1922-00a.
22. RDKit: open-source cheminformatics software. Available from: https://www.rdkit.org. Accessed October 24, 2025.
23. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: machine learning in Python. J Mach Learn Res. 2011; 12:2825–2830.
24. Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, et al. TensorFlow: large-scale machine learning on heterogeneous distributed systems. arXiv. 2016;1603.04467.
25. Hollmann N, Müller S, Eggensperger K, Hutter F. TabPFN: a transformer that solves small tabular classification problems in a second. In: International Conference on Learning Representations; 2023