Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Optimized Gradient Boosting Decision Tree Ensembles for Honeybee Toxicity Prediction Using the ApisTox Dataset

Sedat GOLGİYAZ

sedatg@bingol.edu.tr

Bingöl University

Research Article

Keywords: ApisTox, Honeybee toxicity, Gradient boosting, Ensemble learning, Molecular fingerprints

Posted Date: July 16th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10325586/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License. Read Full License

Additional Declarations: No competing interests reported.

Abstract

Honeybee toxicity prediction is essential for agrochemical risk assessment, yet conventional LD_{50} bioassays are labor-intensive and unsuitable for large-scale screening. This study proposes a meta-feature-enriched Gradient Boosting Decision Tree (GBDT) framework for binary toxicity prediction using the ApisTox dataset. Three molecular representations (ECFP, Avalon, and PhysChem&MolTop) were combined with agrochemical functional class meta-features and used to optimize the CatBoost, LightGBM, and XGBoost models. The hyperparameters were optimized using 500 Optuna trials, and the ensemble strategies were evaluated using time-based and max-min splitting protocols. Meta-features improved the MCC from 0.90 to 0.91 for the time-based split and from 0.58 to 0.66 for the max-min split. Compared with the original ApisTox benchmark, the proposed framework increased the MCC from 0.48 to 0.91 (+ 89.6%) for the time-based split and from 0.49 to 0.66 (+ 34.7%) for the max-min split. These results demonstrate the effectiveness of combining optimized GBDT models, domain-specific meta-features, and ensemble learning for honeybee toxicity prediction.

1. INTRODUCTION

Honeybee toxicity prediction has become an increasingly important research topic owing to the ecological and economic significance of pollinators. Beyond protecting biodiversity, reliable toxicity prediction supports sustainable pesticide development and regulatory risk assessment while reducing dependence on animal experiments. Consequently, computational toxicology has increasingly adopted Quantitative Structure–Activity Relationship (QSAR) modeling as a practical alternative to conventional bioassays. By establishing relationships between molecular structure and biological activity, QSAR models enable the rapid screening of large chemical libraries while substantially reducing experimental costs and time [1]. Recent applications further demonstrate the importance of QSAR for environmental risk assessment of pesticides and emerging contaminants within the framework of OECD recommendations [2].

As molecular datasets have expanded, machine learning (ML) algorithms have gradually replaced conventional statistical QSAR techniques, such as multiple linear regression and support vector machines (SVMs). Traditional descriptor-based approaches often require extensive feature engineering and may fail to capture the highly nonlinear relationships inherent in molecular structures. To overcome these limitations, molecular fingerprints, including Extended Connectivity Fingerprints (ECFP) and Avalon fingerprints, have become the dominant representations for toxicity prediction because they efficiently encode chemically meaningful structural information while remaining computationally efficient [3], [4].

Recently, Graph Neural Networks (GNNs) have introduced a new paradigm in QSAR modeling by directly learning molecular representations from graph structures derived from Simplified Molecular Input Line Entry System (SMILES) strings without requiring handcrafted descriptors [5], [6]. Through message passing mechanisms, GNNs automatically aggregate atomic and bond information to capture complex structural relationships, leading to improved predictive performance for many molecular property prediction tasks [7]–[9]. The Weisfeiler–Lehman Optimal Assignment (WL-OA) graph kernel represents molecules by comparing graph structures through iterative node relabeling and optimal assignment, providing an effective similarity measure for graph-based machine learning models [11]. Nevertheless, their performance remains highly dependent on the dataset size, chemical diversity, and computational resources [10].

A major milestone in honeybee toxicity prediction was the introduction of the ApisTox [12] benchmark, which integrates experimentally validated toxicity data from the ECOTOX, PPDB, and BPDB. ApisTox is a standardized dataset containing 1,035 compounds. By combining acute oral and contact toxicity endpoints and introducing chemically diverse max-min, time, and random data partitions, ApisTox provides a realistic benchmark for evaluating ML models under challenging generalization scenarios [4], [10].

Since the release of ApisTox, only a limited number of studies have used this benchmark. Adamczyk et al. established comprehensive baseline results by comparing molecular fingerprints, graph kernels, GNNs, pretrained molecular models. Classical ML algorithms demonstrate that fingerprint-based approaches combined with tree-based learning remain highly

competitive despite the rapid development of deep learning methods [4]. More recently, Damião et al. developed a GNN-based QSAR model using molecular graphs derived directly from SMILES strings and achieved a Matthews Correlation Coefficient (MCC) of 0.552 [6].

From an ML perspective, improving predictive performance generally relies on four complementary strategies: informative feature engineering or feature selection [13], model optimization [14], hyperparameter tuning [15], and ensemble learning [16]–[18]. Feature engineering aims to enrich the input representation by incorporating chemically meaningful information, whereas feature selection identifies the most discriminative variables while reducing the redundancy and computational complexity. Model optimization further enhances the predictive capability through systematic hyperparameter tuning, and ensemble learning improves robustness by combining the strengths of multiple base learners. Although each of these strategies has demonstrated effectiveness individually, they have rarely been investigated together for honeybee toxicity prediction using the ApisTox Benchmark. In particular, the integration of complementary molecular representations with domain-specific agrochemical information and optimized ensemble learning remains largely unexplored.

To address these limitations, this study proposes a meta-feature-enriched Gradient Boosting Decision Tree (GBDT) ensemble framework for honeybee toxicity prediction. Three complementary molecular representations (ECFP, Avalon, and PhysChem–MolTop descriptors) were combined with agrochemical functional class indicators to enhance feature representation. Optimized CatBoost, LightGBM, and XGBoost models were developed through Bayesian hyperparameter optimization and subsequently integrated using ensemble learning strategies. The proposed framework was evaluated on the ApisTox benchmark under both chemically diverse max-min and time-based data partitions, aiming to improve predictive accuracy, robustness, and generalization while maintaining computational efficiency and model interpretability.

2. MATERIALS AND METHODS

2.1. ApisTox Dataset

Figure 1. summarizes the composition of ApisTox dataset. Herbicides constituted the largest agrochemical group (354 compounds), followed by insecticides (227), other agrochemicals (216), and fungicides (197) (Fig. 1a). The dataset integrates compounds from three curated databases, with PPDB contributing 49.3% of the molecules, followed by ECOTOX (42.6%) and the BPDB (8.1%). Based on the U.S. EPA acute toxicity threshold (LD_{50} ≤ 11 µg/bee), the dataset contained 296 toxic (28.6%) and 739 non-toxic (71.4%) compounds (Fig. 1b). Owing to the moderate class imbalance, the model performance was primarily assessed using the MCC. Synthetic oversampling techniques were intentionally avoided because artificially generated samples may not preserve chemically valid molecular structures within the curated ApisTox benchmark.

To further characterize the dataset, Fig. 2 illustrates the temporal distribution of the compounds across the four agrochemical classes from the 1970s to the 2020s. The publication year obtained from the first PubChem record was used as a proxy for the period when each compound became publicly available. This temporal distribution highlights the importance of time-based data splitting. Unlike randomly partitioned data, the time-based split evaluates the model performance on newly introduced compounds and better reflects real-world deployment scenarios. Across all decades, insecticides consistently exhibited the highest proportion of toxic compounds, whereas herbicides were the least toxic. Fungicides and other agrochemicals exhibited intermediate toxicity rates, which gradually declined over time. This trend demonstrates that both the chemical composition and toxicity profiles of agrochemicals have evolved across different periods.

2.1. Feature Extraction and Molecular Representation

To develop robust predictive models, the chemical compounds were represented using molecular descriptors together with agrochemical class information available in the ApisTox dataset. Four binary-encoded meta-features corresponding to the

primary agrochemical functional classes (herbicide, fungicide, insecticide, and other agrochemical) were incorporated into the input feature set. These meta-features were combined with molecular representations to investigate whether the agrochemical class information could improve the predictive performance of the machine learning models.

In addition to these functional meta-features, microscopic structural information was mathematically extracted directly from the SMILES strings using the open-source RDKit cheminformatics framework [19]. For this structural extraction, four distinct molecular representation methods were employed: ECFP, Avalon fingerprints, and Molecular Topological (MolTop) and physicochemical (PhysChem) descriptors. The ECFP (analogous to Morgan fingerprints) was computed to capture circular topological neighborhoods, which is highly effective for identifying specific toxicophore substructures [3]. Avalon fingerprints were generated to account for both structural connectivity and localized physicochemical properties [19], [20]. Crucially, the PhysChem&MolTop feature set was explicitly defined by extracting five fundamental physicochemical and topological indices using the RDKit descriptor modules: Molecular Weight (MolWt), Heavy Atom Count, Ring Count, Topological Polar Surface Area (TPSA) [21], and Bertz Complexity Index (BertzCT) [22].

2.2. Gradient Boosting Decision Trees

Gradient Boosting Decision Trees (GBDT) methods construct an additive ensemble of weak learners (decision trees) by iteratively minimizing a regularized objective function. At iteration t the general objective function is formulated in Eq. 2.

$$\mathcal{L}^{(t)} = \sum\nolimits_{i=1}^{n} L\left(y_i,\ \widehat{y}_i^{(t-1)} + f_t\left(x_i\right)\right) + \Omega\ \left(f_t\right)$$

1

Where:

- L : Loss function.
- $\widehat{y}_i^{(t-1)}$: Cumulative prediction from all previous iterations $(t-1)$ for the i-th observation.
- $f_t\left(x_i\right)$: The newly added tree f_t prediction for the $i-th$ observation.
- f_t : New tree is added to the model in the $t-th$ iteration.
- Ω : Regularization term penalizing model complexity to prevent overfitting
- y_i : Actual value for the $i-th$ observation.
- n : Total number of observations.
- t : Current iteration number.

The regularization term penalizes model complexity, which is expressed in Eq. 2.

$$\Omega\ \left(f_t\right) = \gamma\,\mathrm{T} + \frac{1}{2}\lambda \sum\nolimits_{j=1}^{T} w_j^2$$

2

Where:

- T : Number of leaves.
- w_j : Weight of leaf j.
- γ and λ : Regularization hyper parameters.

CatBoost sets itself apart by introducing Ordered Boosting, a permutation-driven scheme that mitigates the prediction shift and target leakage inherent in classical gradient boosting. Together with a native Ordered Target Statistics encoding for categorical features, this eliminates the need for manual preprocessing [23]. The use of symmetric (oblivious) decision

trees provides implicit regularization and enables extremely fast inference, making CatBoost particularly effective [24] on datasets containing categorical variables and small-to-medium sample sizes, such as ApisTox.

LightGBM differentiates itself through exclusive feature bundling, which merges mutually exclusive sparse features to reduce the dimensionality without significant information loss. Combined with a leaf-wise (best-first) tree growth strategy instead of level-wise growth, LightGBM achieves substantially faster training speed and lower memory consumption while often yielding higher accuracy on large-scale, high-dimensional data [25], [26].

XGBoost distinguishes itself by employing a second-order Taylor expansion of the loss function, incorporating both gradient and Hessian information to achieve more accurate split-point estimation [27]. Its main advantage lies in its highly optimized system-level implementation, featuring parallelized tree construction, cache-aware access patterns, and sparsity-aware split finding, which collectively enable superior scalability on large, high-dimensional datasets [28].

2.3. Single Models Hyperparameter Optimization

To systematically explore the hyperparameter space and prevent the computational inefficiency of traditional grid search methods, the Optuna framework was employed [15]. For each model, 500 optimization trials were conducted. Optuna relies on the Tree-structured Parzen Estimator as its default sequential model-based optimization algorithm, as formulated in Eq. 3.

$$x_{next} = \underset{x \in X}{\operatorname{argmax}} \frac{k(x)}{r(x)}$$

3

Where:

- x_{next} : The next optimal hyperparameter configuration was selected for evaluation.
- X : The entire predefined hyperparameter search space.
- $k(x)$: Probability density function of hyperparameter configurations that yielded evaluation scores better than a predefined threshold.
- $r(x)$: Probability density function of configurations that yielded evaluation scores worse than the threshold.

The specific hyperparameters tuned using the Optuna framework, along with their defined search space boundaries for model optimization, are listed in Table 1.

To ensure computational efficiency, handle class imbalance robustly, and maintain a fair comparison across the models, specific parameters were kept strictly constant during the optimization process. Smaller *iterations/n_estimators* values performed better on the time-split data, and the search space was narrowed down to 11–333 for these models. For all models, the evaluation metric was set to the Area Under the Curve (AUC), and the random state (or seed) was fixed at 42 to guarantee the complete reproducibility of the experimental results. Furthermore, to address the inherent class disparity within the dataset, class weights were configured as 'balanced' for both CatBoost and LightGBM, whereas for XGBoost, the scale_pos_weight parameter was utilized by calculating it dynamically based on the dataset's exact ratio of negative to positive instances.

Table 1
Hyperparameter search space defined for the model optimization.

Model	Hyperparameter	Type	Search Space	Scale
CatBoost	iterations	Integer	[300, 3000]/[11,333]	Linear
	learning_rate	Float	[1e-4, 0.1]	Log
	depth	Integer	[3, 9]	Linear
	l2_leaf_reg	Float	[1e-3, 33.0]	Log
	border_count	Integer	[1, 255]	Linear
	random_strength	Float	[1e-3, 10.0]	Log
	bagging_temperature	Float	[0.0, 1.0]	Linear
	colsample_bylevel	Float	[0.1, 1.0]	Linear
XGBoost	n_estimators	Integer	[300, 3000]]/[11,333]	Linear
	learning_rate	Float	[1e-4, 0.1]	Log
	max_depth	Integer	[3, 9]	Linear
	reg_lambda	Float	[1e-3, 33.0]	Log
	subsample	Float	[0.5, 1.0]	Linear
	colsample_bytree	Float	[0.5, 1.0]	Linear
LightGBM	n_estimators	Integer	[300, 3000]]/[11,333]	Linear
	learning_rate	Float	[1e-4, 0.1]	Log
	max_depth	Integer	[3, 9]	Linear
	num_leaves	Integer	[20, 150]	Linear
	reg_lambda	Float	[1e-3, 33.0]	Log
	subsample	Float	[0.5, 1.0]	Linear
	colsample_bytree	Float	[0.5, 1.0]	Linear

2.4. Ensemble Learning Strategies

To exploit the complementary strengths of CatBoost, LightGBM, and XGBoost, four ensemble schemes were implemented in both pairwise and triplet configurations: soft voting with simple averaging (Eq. 4), soft voting with weighted averaging (Eq. 5), hard voting (Eq. 6), and stacking (Eq. 7).

$$\widehat{P}_{ensemble}\left(y=c|x\right)=\frac{1}{M}\sum\nolimits_{m=1}^{M}P_m(y=c\left|x\right)$$

4

$$\widehat{P}_{ensemble}\left(y=c|x\right)=\frac{\sum\nolimits_{m=1}^{M}w_mP_m(y=c\left|x\right)}{\sum\nolimits_{m=1}^{M}w_m}$$

5

$$\hat{y}_{ensemble}=mode\left\{\hat{y}_1,\hat{y}_2,\ldots,\hat{y}_M\right\}$$

6

$$\hat{y}_{ensemble} = g\left(P_1\ (y|x)\,,\ P_2\ (y|x)\,,\ldots\,,\ P_M\ (y|x)\,;\ \theta\right)$$

7

Where:

- M : Number of base classifiers in ensemble.
- $P_m(y = c \mid x)$: Predicted probability of class $c$by base classifier $m.$
- w_m : Weight assigned to base classifier $m.$
- $\hat{y}_M$: Discrete class prediction of base classifier $m.$
- c : A set of possible class labels.
- $g(.)$: Meta-learner function for stacking.
- θ : Meta-learner parameters.

In the simple soft voting scheme (Eq. 4), the ensemble output is obtained as the unweighted mean of the predicted class probabilities across all base classifiers, providing a baseline-probabilistic aggregation strategy [29]. The weighted soft voting scheme (Eq. 5) generalizes this formulation by assigning each base learner a weight proportional to its model performance, thereby emphasizing more reliable classifiers within the aggregation [30]. In hard voting (Eq. 6), each base classifier casts a discrete class prediction, and the ensemble label is determined by a majority vote, which is particularly robust when base learners exhibit uncorrelated errors [29]. Finally, the stacking scheme (Eq. 7) employs a meta-learner $g(\cdot)$ parameterized by θ , trained on model predictions of the base learners to learn an optimal non-linear combination of their outputs, thereby mitigating information leakage and capturing higher-order dependencies among base models [31].

2.5. Model Evaluation Metrics

The dataset was trained on the training subset and evaluated on the test subset, with four metrics being computed: MCC, Area Under the Receiver Operating Characteristic curve (AUROC), macro-averaged recall, and macro-averaged precision [32]–[34]. The classification performance and robustness of the proposed model against class imbalance were comprehensively evaluated using the MCC (Eq. 8) to assess the overall prediction consistency, macro-averaged precision (Eq. 9), and recall (Eq. 10) to ensure equitable class-wise evaluation, and the Area Under the Receiver Operating Characteristic curve (AUROC, Eq. 11) to capture the overall discriminative capacity of the model.

$$MCC = \frac{TP \times\ TN - FP \times\ FN}{\sqrt{(TP + FP)\,(TP + FN)\,(TN + FP)\,(TN + FN)}}$$

8

$$Precision_{macro} = \frac{1}{N}\sum\nolimits_{i=1}^{N}\frac{TP_i}{TP_i + FP_i}$$

9

$$Recall_{macro} = \frac{1}{N}\sum\nolimits_{i=1}^{N}\frac{TP_i}{TP_i + FN_i}$$

10

$$AUROC = \int_0^1 TPR\left(FPR^{-1}\right)(t)\,dt$$

11

Where:

- TP: True Positive
- TN: True Negative
- FP: False Positive
- FN: False Negative
- TPR: True Positive Rate
- FPR: False Positive Rate

For model evaluation, the predefined splits provided by the ApisTox dataset were utilized, with the max-min split being prioritized, as it employs the maximum diversity picking algorithm with ECFP fingerprints and dice distance to maximize chemical space coverage in the test set. This ensures robust in-domain generalization while minimizing data leakage from structurally similar training compounds. Complementarily, the time-split, constructed by placing the newest molecules (based on PubChem first publication year) into the test set, was employed to assess out-of-domain extrapolative performance, simulating a realistic scenario of predicting toxicity for structurally novel, future agrochemicals.

3. EXPERIMENTAL RESULTS

3.1. Predictive Performance of Optimized Singel Models

The optimal hyperparameter configurations for the three machine learning models trained across three distinct molecular representation spaces (ECFP, Avalon, and PhysChem&MolTop) were determined through 500 Optuna optimization trials aimed at maximizing the MCC. The best parameter sets for the time-based data split are summarized in Table 2.

Table 2
Optimal hyperparameter configurations for time-split dataset.

Hyper parameter	Catboost			LightGBM			XGBoost		
	AVA-LON	ECFP	PChm&MTOP	AVA-LON	ECFP	PChm&MTOP	AVA-LON	ECFP	PChm&MTOP
iterations/ n_estimators	289	198	67	236	215	169	307	129	320
learning_rate	0.0294	0.0185	0.0042	0.0161	0.0491	0.0079	0.0451	0.0996	0.0003
depth /max_depth	8	9	8	9	8	3	3	9	3
num_leaves	–	–	–	126	136	61	–	–	–
l2_leaf_reg/ reg_lambda	21.3193	0.0092	0.1395	0.0021	9.1753	0.2068	32.6577	18.6974	0.0297
border_count/ scale_pos_weight	126	220	185	–	–	–	2.2471	2.2471	2.2471
random_strength	0.0072	0.7697	0.0297	–	–	–	–	–	–
bagging_temperature	0.4306	0.5380	0.9189	–	–	–	–	–	–
subsample	–	–	–	0.7048	0.6074	0.5559	0.8974	0.5270	0.8022
colsample_bylevel /colsample_bytree	0.2232	0.4589	0.3469	0.5126	0.5472	0.8646	0.9322	0.7964	0.9726

Similarly, the optimal hyperparameter configurations for the three GBDT models across the three molecular representation spaces were determined using the max-min data-splitting strategy. The best parameter sets, obtained through 500 independent Optuna optimization trials aimed at maximizing the MCC, are listed in Table 3.

After the hyperparameter optimization process, the predictive capabilities of the three models across the three feature representation spaces were evaluated using the best parameter sets obtained from the Optuna trials. The resulting performance metrics were visualized as heatmaps to facilitate a comprehensive comparison. The evaluation results for the time-based data split are presented in Fig. 3, followed by the results for the max-min data-splitting strategy in Fig. 4.

Table 3
Optimal hyperparameter configurations for the max-min split dataset.

Hyperparameter	Catboost			LightGBM			XGBoost		
	AVA-LON	ECFP	PChm&MTOP	AVA-LON	ECFP	PChm&MTOP	AVA-LON	ECFP	PChm&MTOP
iterations / n_estimators	2029	1736	1280	857	1630	1633	380	2097	1596
learning_rate	0.0179	0.0011	0.0004	0.0378	0.0011	0.0876	0.0016	0.0028	0.0141
depth / max_depth	6	8	5	9	3	7	9	3	7
num_leaves	–	–	–	134	75	49	–	–	–
l2_leaf_reg / reg_lambda	0.0092	4.0426	32.9052	0.0057	0.2075	0.6692	0.0068	0.0908	0.0347
border_count / scale_pos_weight	132	203	250	–	–	–	2.2598	2.2598	2.2598
random_strength	1.9574	4.5854	1.2811	–	–	–	–	–	–
bagging_temperature	0.3280	0.0917	0.0359	–	–	–	–	–	–
subsample	–	–	–	0.5003	0.6760	0.7458	0.9446	0.6767	0.8013
colsample_bylevel / colsample_bytree	0.5561	0.1549	0.8506	0.7334	0.8419	0.5410	0.5475	0.6486	0.6268

Figure 3 shows the results of the time-based splitting strategy. The XGBoost model paired with ECFP features demonstrated absolute superiority. This combination achieved the highest scores across all evaluated metrics (MCC, AUROC, Precision, and Recall). This specific split requires the prediction of temporally novel compounds. It strictly tests out of distribution generalization. The results indicate that topological data from ECFP synergize optimally with the XGBoost architecture in this scenario.

Figure 4 presents the results for the max-min data split. This strategy maximized the structural diversity of the test set. Here, the leading performance was distributed among the different models. The XGBoost and AVALON combination exhibited the highest overall classification power. It secured the highest MCC and Precision scores.

A comparison of the two strategies revealed significant behavioral differences between the models. The time-based split introduces a highly challenging prediction scenario (Fig. 3). Models must evaluate entirely new molecules. Under these conditions, predictive success depends heavily on a specific feature (ECFP) and the model (XGBoost).

Conversely, the max-min split tests the structural boundaries of the known chemical space (Fig. 4). In this environment, the AVALON fingerprint yielded more stable and consistent outcomes. It performed particularly well alongside tree-based algorithms like XGBoost and CatBoost. This sharp variation in performance highlights a critical conclusion: the chosen dataset splitting methodology directly dictates the optimal model and feature selection in molecular machine learning.

3.2. Ensemble Model Results

To further enhance the predictive performance, various ensemble learning techniques were evaluated specifically for time-based and max-min data splits. Both voting and stacking methodologies were systematically explored. The base models consisted of two- and three-model combinations of CatBoost, LightGBM, and XGBoost. For the voting ensembles, three distinct strategies were applied: hard voting (majority rule), simple average soft voting, and weighted soft voting. Stacking architectures were also developed using the same base model combinations. Several algorithms were tested as the final

meta-learner to aggregate base predictions. The final estimators included Logistic Regression, Extra Trees, CatBoost, XGBoost, and LightGBM. This comprehensive approach aimed to maximize overall classification robustness by combining individual algorithmic strengths. The performance metrics achieved by the top ten ensemble configurations are presented in Fig. 5.

Figure 5 highlights the top ten performing ensemble configurations for the time-based data split. The results demonstrate that the stacking architecture consistently outperforms the voting strategy. Specifically, the combination of ECFP features with base models, such as CatBoost and XGBoost, yielded the highest predictive power. The Stacking approach using LGBM as the final meta-learner achieved a peak MCC score of 0.9104. This confirms that aggregating predictions using a robust meta-learner effectively mitigates the limitations of the individual base models. Furthermore, the performance gap between the top-ranked models and the remaining ensemble configurations was relatively narrow. This suggests that the ECFP feature space is highly compatible with diverse ensemble methods in this temporal scenario. Overall, these results validate that ensemble learning successfully enhances classification robustness, even when dealing with structurally novel compounds.

Figure 6 illustrates the top 10 ensemble performances using the max-min splitting strategy. Unlike the time split, the AVALON feature representations dominated the leaderboard in this scenario. Voting strategies, particularly hard voting, demonstrate unexpected effectiveness alongside stacking architectures.

For the Max-Min split, the AVALON-based LightGBM and XGBoost ensembles achieved the highest MCC of 0.6663 using hard voting. However, this performance was nearly identical to that of the best individual model, indicating that ensemble learning provided only a marginal benefit under maximum structural diversity. These results suggest that well-optimized single GBDT models are already sufficient under this evaluation scenario.

In the time-based split (Fig. 5), the ECFP-based representations consistently achieved the highest predictive performance. This evaluation protocol assesses the model performance of compounds introduced after the training period and therefore provides a more realistic estimate of prospective prediction. In contrast, the max-min split (Fig. 6), which maximizes the structural diversity between the training and test sets, favored AVALON-based representations. These results indicate that the relative performance of molecular representations depends on the evaluation protocol and structural characteristics of the dataset.

In contrast, under the max-min split, ensemble learning did not improve the predictive performance over the best individual model. The best hard voting ensemble achieved the same MCC (0.6663) as the strongest single GBDT model, indicating that well-optimized individual models were already sufficient for maximum structural diversity.

4. CONCLUSION

The comparative results in Table 4 demonstrate that incorporating agrochemical functional class information as meta-features consistently improved the performance of the proposed framework. Under the time-based split, the MCC increased from 0.90 using ECFP fingerprints to 0.91 after adding the meta-features. A larger improvement was observed under the max-min split, where the MCC increased from 0.58 to 0.66, using the Avalon representation. These results indicate that agrochemical class information provides additional predictive value, particularly in chemically diverse evaluation scenarios.

Table 4
Performance comparison between the proposed framework and previously published ML models on the ApisTox benchmark.

Split Metod	Features Metod	Model	MCC/ OOF-MCC	AVG-MCC	AUROC/ OOF-AUROC	Precision/ OOF-Precision	Recall/ OOF-Recall
Time	ECFP[4]	RF	0.48	-	78.16%	72.95%	42.10%
MinMax	Avalon[4]	RF	0.48	-	76.09%	76.37%	39.67%
Time	WL-OA[4]	SVM	0.43	-	78.08%	62.07%	43.90%
MinMax	WL-OA[4]	SVM	0.49	-	83.95%	62.16%	54.76%
Time	Meta + ECFP	Proposed	**0.91**	**-**	**98.07%**	**95.5%**	**95.52%**
MinMax	Meta + AVALON	Proposed	**0.66**	**-**	**83.84%**	**69.97%**	**78.35%**
Time	ECFP	Proposed	**0.90**	**-**	**92.24%**	**97.2%**	**93.74%**
MinMax	AVALON	Proposed	**0.58**	**-**	**80.03%**	**90.99%**	**71.13%**
5-Fold	MOLECULAR-GRAPHS[6]	GNNs	-	0.552	75.9%	72.3%	62.5%
5-Fold	Meta + AVALON	Proposed	**0.87**	**0.87**	**96.93%**	**93.46%**	**93.54%**

To evaluate the contribution of model optimization, the proposed GBDT models were also assessed using molecular fingerprints alone (Table 4). Even without meta-features, the optimized models achieved strong performance, with MCC values of 0.90 for the time-based split and 0.58 for the max-min split. This demonstrates that systematic Bayesian hyperparameter optimization substantially improves the predictive performance, whereas meta-features provide an additional gain.

To compare with previous studies, the proposed framework was evaluated using the same protocols reported in the literature (Table 4). Under five-fold cross-validation, the recently published GNN model achieved an average MCC of 0.552, whereas the proposed framework reached 0.87, corresponding to an improvement of 57.6% [6]. For the time-based split, the original ApisTox benchmark reported an MCC of 0.48 using Random Forest with ECFP fingerprints. The optimized fingerprint-only model increased the MCC to 0.90 (+ 87.5%), while the addition of meta-features further increased it to 0.91 (+ 89.6%). Under the Max-Min split, the best benchmark result was an MCC of 0.49, obtained using the WL-OA graph kernel with an SVM. The optimized Avalon model achieved an MCC of 0.58 (+ 18.4%), which further increased to 0.66 (+ 34.7%) after incorporating meta-features.

Overall, the proposed framework improved the predictive performance through three complementary components: optimized GBDT models, agrochemical meta-features, and ensemble learning. Model optimization produced the largest performance improvement over previous studies. Meta-features further improved the performance, particularly under the max-min split. Ensemble learning consistently achieved the best overall results under the time-based split, whereas under the max-min split, it matched the performance of the best individual model. These findings demonstrate that combining optimized GBDT models with domain-specific meta-features and appropriate ensemble strategies provides an accurate and robust framework for honeybee toxicity prediction using curated molecular datasets.

Declarations

Author Contribution

S.G. conceived the study, designed the methodology, performed all experiments, analyzed the results, and wrote the manuscript. The author reviewed and approved the final version of the manuscript.

Data Availability

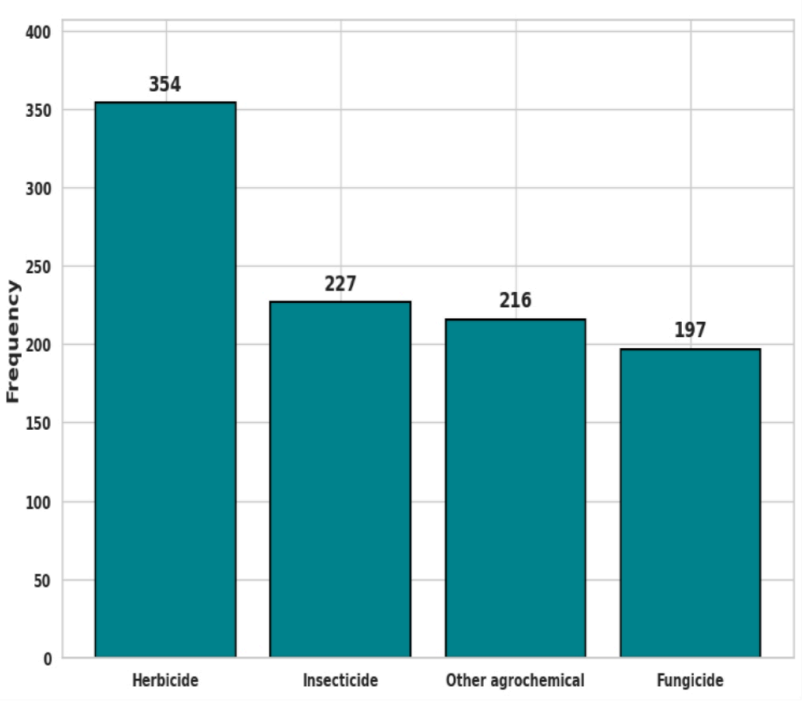
The dataset analyzed in this study is publicly available through the ApisTox benchmark. All data used in this work can be obtained from the original publication and its associated repository(10.5281/ZENODO.13350981**).**

References

1. Cherkasov A et al (2014) QSAR Modeling: Where Have You Been? Where Are You Going To? *J. Med. Chem.*, vol. 57, no. 12, pp. 4977–5010, Jun. 10.1021/jm4004285
2. Kumar A, Ojha PK, Roy K (Dec. 2024) First report on regression-based QSAR addressing pesticide dissipation half-life in plants: A step towards sustainable public health. Sci Total Environ 954:176175. 10.1016/j.scitotenv.2024.176175
3. Rogers D, Hahn M, Fingerprints E-C (May 2010) J Chem Inf Model 50(5):742–754. 10.1021/ci100050t
4. Adamczyk J, Poziemski J, Siedlecki P (Mar. 2026) Evaluating machine learning models for predicting pesticide toxicity to honeybees. Ecotoxicol Environ Saf 312:119869. 10.1016/j.ecoenv.2026.119869
5. Duvenaud RP, Maclaurin DK, Iparraguirre D, Bombarell J, Hirzel R, Aspuru-Guzik T, A., Adams (2015) Convolutional networks on graphs for learning molecular fingerprints, in *Advances in neural information processing systems*, p. 28. 10.5555/2969442.2969488
6. Damião TC, de Oliveira Neto RF, de Filho EB (2025) New generation of QSAR modeling for bee safety: predicting toxicity using graph neural networks and apistox data, *Ecotoxicology*, vol. 34, no. 10, pp. 2028–2039, Dec. 10.1007/s10646-025-02970-0
7. Zhou J et al (2020) Graph neural networks: A review of methods and applications. AI Open 1:57–81. 10.1016/j.aiopen.2021.01.001
8. Hung C, Gini G (Aug. 2021) QSAR modeling without descriptors using graph convolutional neural networks: the case of mutagenicity prediction. Mol Divers 25(3):1283–1299. 10.1007/s11030-021-10250-2
9. Ketkar R, Liu Y, Wang H, Tian H (Jul. 2023) A Benchmark Study of Graph Models for Molecular Acute Toxicity Prediction. Int J Mol Sci 24(15):11966. 10.3390/ijms241511966
10. Adamczyk J, Poziemski J, Siedlecki P (Jan. 2025) ApisTox: a new benchmark dataset for the classification of small molecules toxicity on honeybees. Sci Data 12(1). 10.1038/s41597-024-04232-w
11. Kriege NM, Giscard P-L, Wilson RC (2017) On Valid Optimal Assignment Kernels and Applications to Graph Classification, Jan. [Online]. Available: http://arxiv.org/abs/1606.01141
12. Adamczyk J, Poziemski J, Siedlecki P ApisTox, 10.5281/ZENODO.13350981
13. Hamdard APMS, Lodin APH (Sep. 2023) Effect of Feature Selection on the Accuracy of Machine Learning Model. Int J Multidiscip Res Anal 06(09). 10.47191/ijmra/v6-i9-66
14. Karthick K (Feb. 2024) Comprehensive Overview of Optimization Techniques in Machine Learning Training. Control Syst Optim Lett 2(1):23–27. 10.59247/csol.v2i1.69
15. Akiba T, Sano S, Yanase T, Ohta T, Koyama M (2019) Optuna, in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2623–2631. 10.1145/3292500.3330701
16. Patel P, Evaluating Ensemble Learning Strategies for Enhanced Medical Diagnostics: Insights from Real-World Datasets, in (2025) *6th International Conference on Problems of Cybernetics and* Informatics *(PCI)*, Aug. 2025, pp. 1–4. 10.1109/PCI66488.2025.11219757
17. sherif samir sultan M, Ewieda A, Gomaa, Yahia MF (2025) Enhancing Classification Accuracy Using Stacked Ensemble Learning: A Hybrid Ensemble Strategy, *ALRYADA J. Comput. Intell. Technol.*, vol. 2, no. 1, pp. 1–24, Dec. 10.21608/ajcit.2025.394368.1014

18. Shah M, Gandhi K, Patel KA, Kantawala H, Patel R, Kothari A, Theoretical Evaluation of Ensemble Machine Learning Techniques, in (2023) *5th International Conference on Smart Systems and Inventive* Technology *(ICSSIT)*, Jan. 2023, pp. 829–837. 10.1109/ICSSIT55814.2023.10061139
19. RDKit (2026) http://www.rdkit.org/ (accessed Jul 09
20. Gedeck P, Rohde B, Bartels C (2006) QSAR – How Good Is It in Practice? Comparison of Descriptor Sets on an Unbiased Cross Section of Corporate Data Sets, *J. Chem. Inf. Model.*, vol. 46, no. 5, pp. 1924–1936, Sep. 10.1021/ci050413p
21. Ertl P, Rohde B, Selzer P (2000) Fast Calculation of Molecular Polar Surface Area as a Sum of Fragment-Based Contributions and Its Application to the Prediction of Drug Transport Properties, *J. Med. Chem.*, vol. 43, no. 20, pp. 3714–3717, Oct. 10.1021/jm000942e
22. Bertz SH (1981) The first general index of molecular complexity, *J. Am. Chem. Soc.*, vol. 103, no. 12, pp. 3599–3601, Jun. 10.1021/ja00402a071
23. Dorogush AV, Ershov V, Gulin A (2018) CatBoost: gradient boosting with categorical features support, [Online]. Available: http://arxiv.org/abs/1810.11363
24. Prokhorenkova L, Gusev G, Vorobev A, Dorogush AV, Gulin A (2019) CatBoost: unbiased boosting with categorical features, Jan. [Online]. Available: http://arxiv.org/abs/1706.09516
25. Ke G et al LightGBM: A Highly Efficient Gradient Boosting Decision Tree.
26. Guolin Ke T-YL, Meng Q, Finley T, Wang T, Chen W, Ma W, Ye Q LightGBM: A Highly Efficient Gradient Boosting Decision Tree, 2017, [Online]. Available: https://proceedings.neurips.cc/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html
27. Chen T, Guestrin C (2016) XGBoost: A Scalable Tree Boosting System. Jun. 10.1145/2939672.2939785
28. Mamani Lopez JW, Morales Gonzales AV, Chambi Condori PP (2025) Predictive models based on machine learning to analyze the adoption of digital payments in Latin America and the Caribbean. Int J Data Netw Sci 9(3):411–418. 10.5267/j.ijdns.2025.3.001
29. Kittler J, Hatef M, Duin RPW, Matas J (1998) On combining classifiers, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 3, pp. 226–239, Mar. 10.1109/34.667881
30. Kuncheva LI (2014) Combining Pattern Classifiers. Wiley
31. Wolpert DH (1992) Stacked generalization. Neural Netw 5(2):241–259. 10.1016/S0893-6080(05)80023-1
32. Chen P, Ye J, Chen G, Zhao J, Heng P-A (2021) Robustness of Accuracy Metric and its Inspirations in Learning with Noisy Labels, *Proc. AAAI Conf. Artif. Intell.*, vol. 35, no. 13, pp. 11451–11461, May 10.1609/aaai.v35i13.17364
33. Sokolova M, Lapalme G (2009) A systematic analysis of performance measures for classification tasks. Inf Process Manag 45(4):427–437. 10.1016/j.ipm.2009.03.002
34. Matthews BW (Oct. 1975) Comparison of the predicted and observed secondary structure of T4 phage lysozyme. Biochim Biophys Acta - Protein Struct 405(2):442–451. 10.1016/0005-2795(75)90109-9

Figures



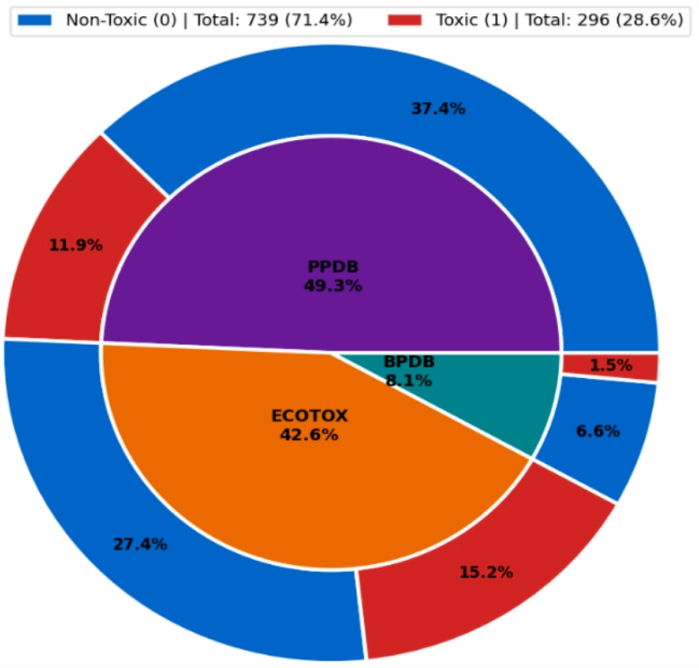


Figure 1

Overview of the ApisTox dataset. (a) Distribution of compounds across agrochemical classes. (b) Contributions of the three source databases (ECOTOX, PPDB, and BPDB) to the toxic and non-toxic classes.

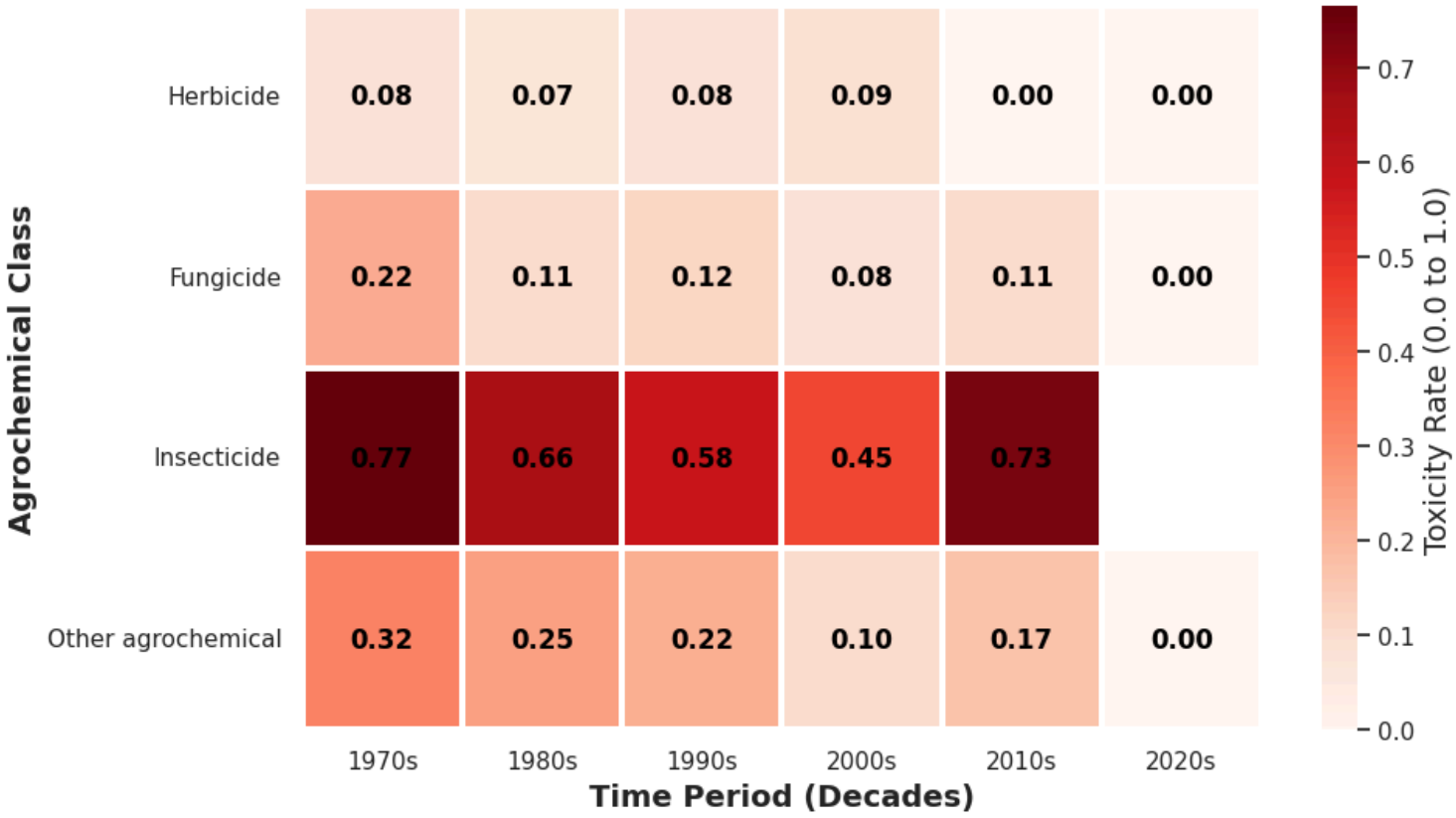


Figure 2

Temporal distribution of honeybee toxicity across agrochemical classes (1970s–2020s).

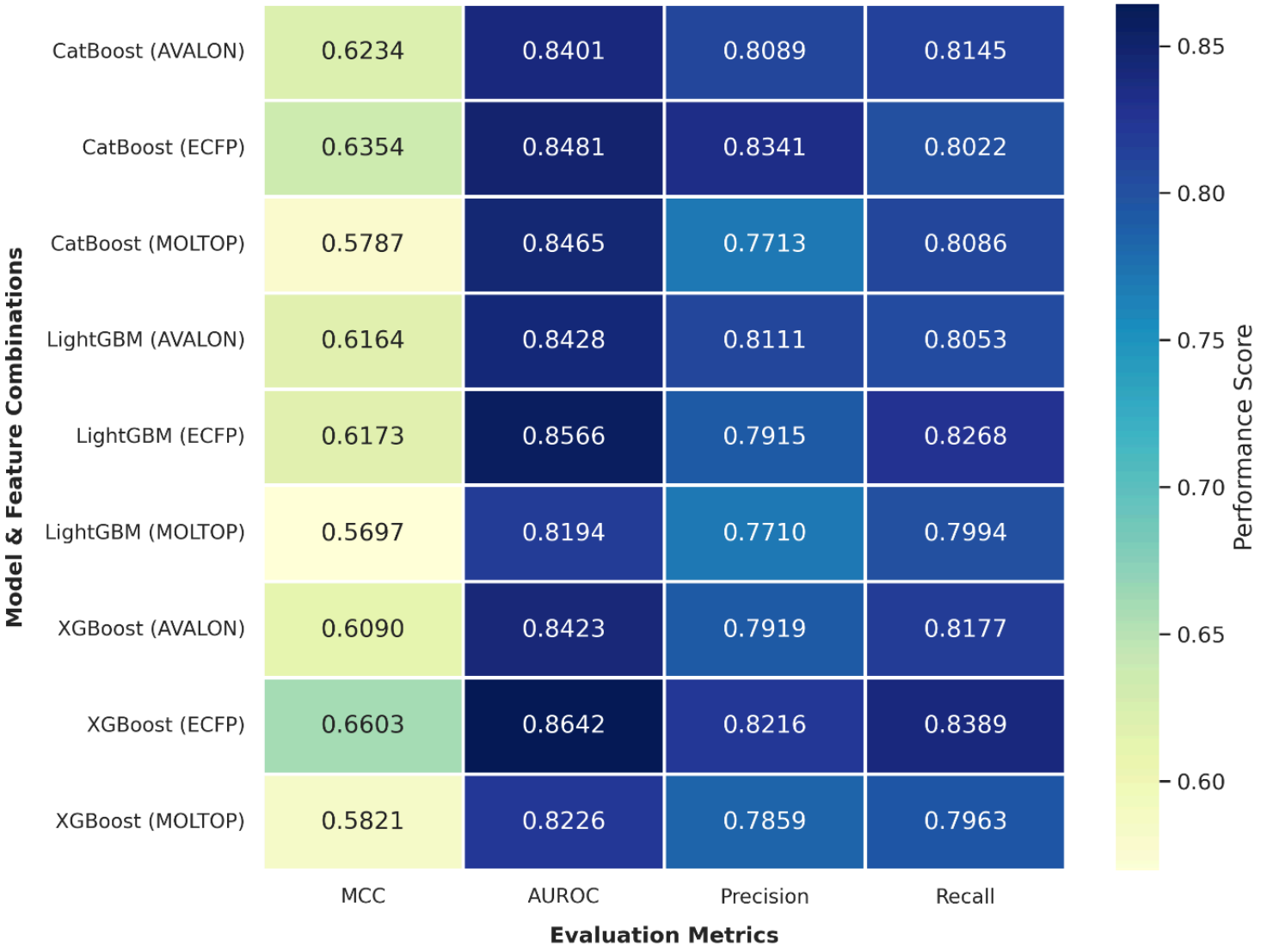


Figure 3

Heatmap of predictive performance metrics across optimized GBDT models and feature representations for the time-based data split.

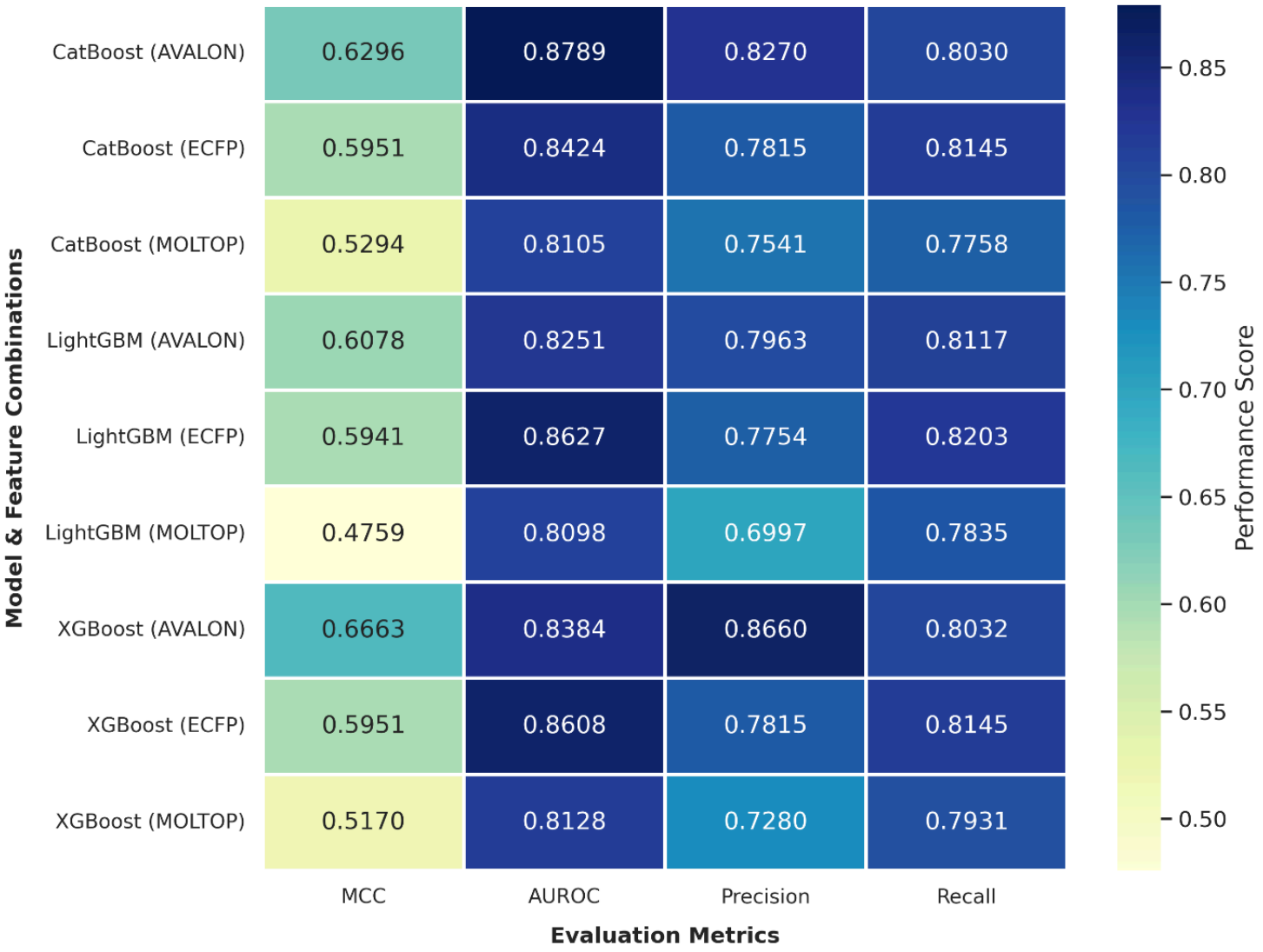


Figure 4

Heatmap of predictive performance metrics across optimized models and feature representations for the max-min data split.

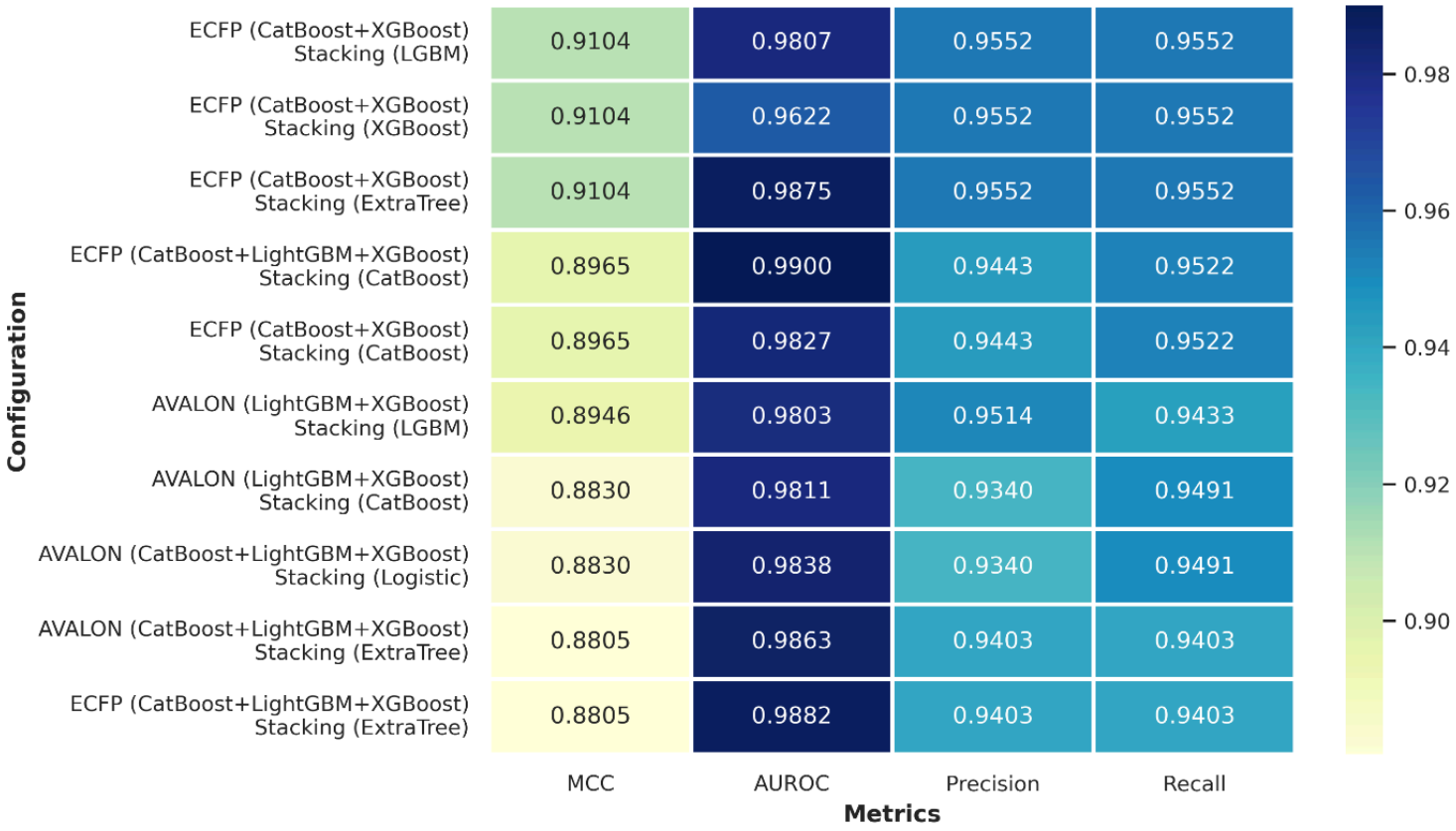


Figure 5

Predictive performance of the top 10 ensemble models for time-based data split.

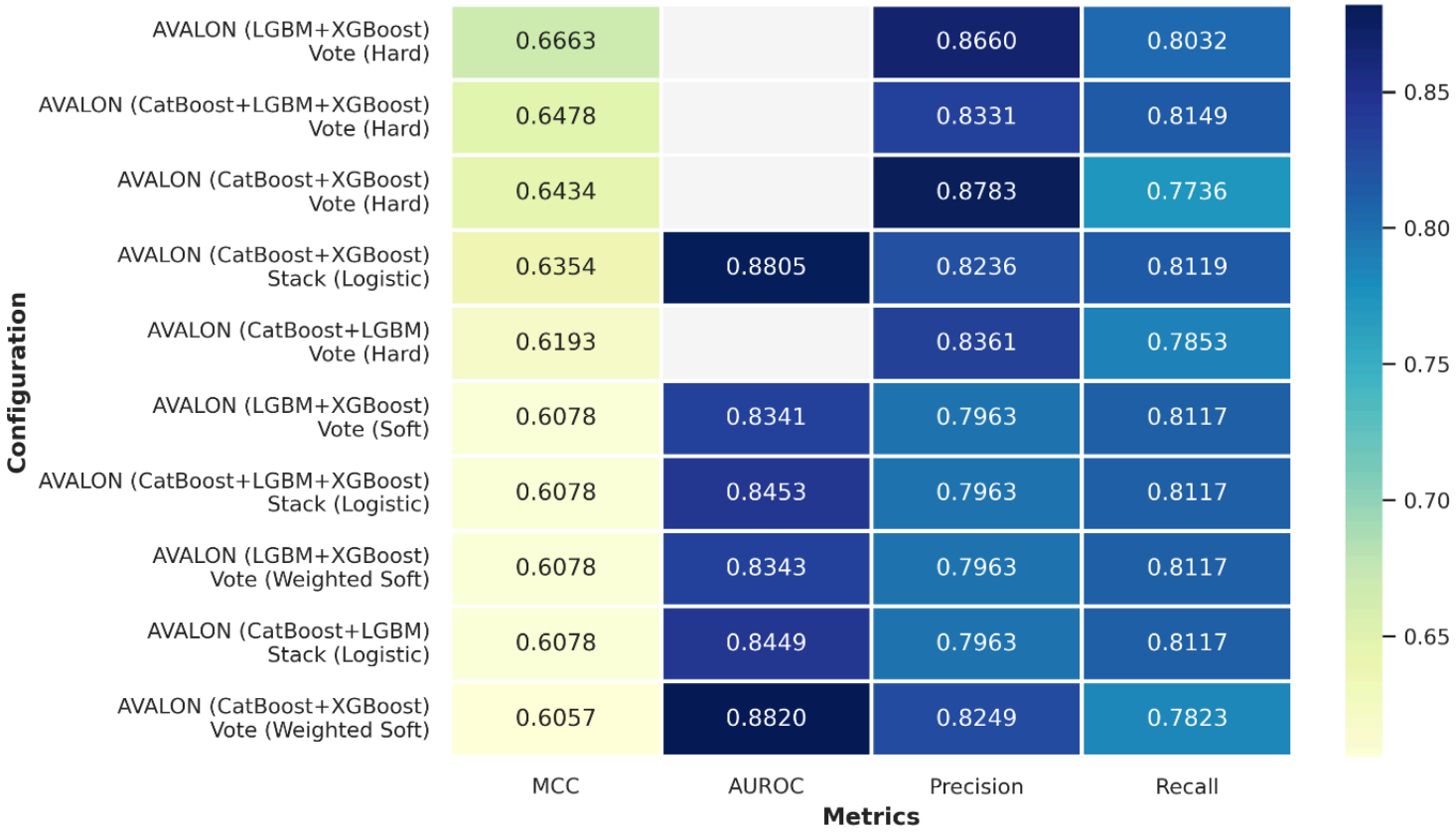


Figure 6

Predictive performance of the top 10 ensemble models for the max-min data split.