

Ensemble Machine Learning Models for Multi-Target Nanoparticle Cytotoxicity Prediction

Abstract— Nanoparticle cytotoxicity prediction from physicochemical descriptors, exposure descriptors, and biological response descriptors is extended with a reproducible machine learning ensemble pipeline, which essentially includes nanoparticle descriptor elaboration and assessment of biologically relevant information. A strong imbalance among classes in the three-class toxicity labels defines the benchmarking of strong tree-based learners with a stacked ensemble that combines Random Forest and LightGBM with a Logistic Regression meta-learner from a curated dataset of 308 nanoparticle records with 22 engineered features. We lay heavy emphasis on ensuring trustworthy evaluation by leakage auditing and stratified splitting so that robust cross-validation behavior and a frozen versioned final model with confidence intervals are reported. The Frozen stacked ensemble scored a macro-F1 of 0.959 and balanced accuracy of 0.944 on a leakage-audited held-out test split, contrasting with five-fold evaluations of baseline models that achieved macro-F1 ≈ 0.829 and balanced accuracy ≈ 0.833 , altogether reflecting the effect of class imbalance and split realism. Besides accuracy, the pipeline also produces publication-ready interpretability outputs (global feature importance and case-level explanations) that would certainly support safer-by-design screening and decision support.

Keywords— Nanoparticle Cytotoxicity Prediction, Physicochemical Descriptors, Ensemble Learning, Stacked Model, Leakage Auditing, Interpretability.

I. INTRODUCTION

Engineered nanoparticles (NPs) are being adopted for biomedical, environmentally friendly, and other applications in the industry; nevertheless, cytotoxic effects could vary widely from material to material, from exposure conditions, and from one biological assay to another. The traditional means of toxicity assessment are focused largely on laboratory testing, and this is an expensive, time-consuming process, which is being made even more difficult to scale within the rapidly growing design space where nanoparticle compositions and formulations are concerned. Thus, an avenue of predictive modeling for initial screening activities and development under safer-by-design paradigms has presented itself, provided that such models are reproducible and tested free of hidden data leakage. There are three reasons why predicting the cytotoxicity of nanoparticles is a challenge. First, toxicity is defined by the n -way interaction of factors along the axes of physicochemical properties (size-related behavior, surface reactivity proxies), exposure descriptors, and biological response descriptors; effective models have to bundle all these heterogeneous signals in a manner that avoids their overfitting. In the second place, datasets dealing with cytotoxicity tend to be strongly imbalanced, where the high-toxicity outcomes are comparatively rarer, thus making naive learning targets inflate apparent performance levels while, instead, failing on minority classes. Thirdly, nanoparticle datasets often contain repeated or correlated records (the same nanoparticle measured under multiple assays), which may cause optimistic evaluation unless one takes special care to

prevent leakage during these splits. The authors present a reproducible ensemble machine-learning pipeline for three-class nanoparticle cytotoxicity classification based on a curated dataset of 308 nanoparticle records with 22 engineered features. We benchmark strong tree-based learners under imbalance-aware training and then construct a stacked-ensemble combining Random Forest and LightGBM with a Logistic Regression meta-learner. In order to draw trustworthy conclusions, we have put emphasis on leakage auditing and stratified splitting, with reporting of robust cross-validation behaviour alongside the provision of the frozen, versioned final model with confidence intervals. Beyond accuracy, the pipeline provides publishable interpretability output—global feature influence and case-based explanations—to assist in screening decisions and mechanistic plausibility checks. The resultant system is therefore placed as a pragmatic and auditable step towards a reliable computational approach to nanotoxicity assessment and risk prioritization that would meet regulatory expectations.

II. RELATED WORKS

The early nano-QSAR studies, using machine learning to predict nanoparticle (NP) cytotoxicity, have developed into more defined nanoinformatics pipelines designed for scalable screening and safe-by-design development. Early nano-QSAR showed that cytotoxicity of metal/metal-oxide NPs could be predicted by correlating it to other known physicochemical descriptors in models that provide statistically meaningful prediction performance. Such both established the feasibility and evinced challenges associated with limited sample sizes and varied endpoints [1], [2]. Subsequent reviews emphasized that a successful model would derive substantively from descriptor quality (e.g., stability, surface-related proxies), consensus experimental context, and transparent reporting related to applicability domains and not by the learning algorithm alone [3], [4]. According to these surveys, the predominating factor for a transferability barrier is heterogeneity in assay conditions and label definitions and makes data measurement and evaluation explicit via reproducible workflows [3], [4].

In addition to modeling improvements, nanoinformatics research ascribes important consideration to the interoperable representation of data and the FAIR architecture. Efforts made to standardize nanomaterials description and metadata have primarily focused on reducing ambiguities surrounding the possible ways by which materials, exposures, and biological contexts can be described or encoded for predictive modeling [4]. Databases and platforms such as eNanoMapper, as well as bigger knowledge-base initiatives (e.g., NanoCommons), were begun to consolidate nanosafety data and hold them for reuse while exposing the current gaps in harmonization and completeness among sources [5]–[7]. It further asserts that nanosafety datasets continue to exist as fragmented stores, lacking critical metadata fields, and often poorly structured for strong ML, especially when external validation is implied [8],[9]. These evidence the need for modeling

pipelines that maintain track of preprocessing choices and keep auditable dataset versions.

Actually, in the algorithmic aspect, cytotoxicity predictors these days increasingly appear to prefer an ensemble learning framework and workflow automation for tabular descriptors. An example of the biologically relevant descriptor that has been shown to be beneficial influence the potential improvements of screening beyond simply physicochemical utilities is the incorporation of biological context such as protein corona fingerprints [10]. Explanatory or so-called "translucent" ML approaches would appear to offer the advantage of joining powerful predictors with interpretable rationales that would support mechanistic plausibility checks and decision support [11]. Systematic preprocessing, selection of algorithm, and hyper-parameter optimization yield important improvements, as shown by comparative AutoML and benchmarking studies, although the end results are very much sensitive to curing decisions and the design of evaluation [12]. Thus, the multi-target nano-QSAR systems intend to maximize applicability by predicting cytotoxicity across several cell lines or some materials as it adds to the growing interest of generalization beyond narrowly scoped datasets [13].

III. SYSTEM OVERVIEW

Table I gives a brief about proposed system for the proposed cytotoxicity prediction pipeline.

Table I: System Overview

Stage	Purpose	Minimal Inputs	Core Method (Low-Parameters)	Primary Outputs
Data Intake & QC	Load + validate dataset	308 records; toxicity labels	schema check, missingness check	cleaned dataset
Feature Matrix	Prepare modeling-ready inputs	22 features (engineered); 3 classes	encoding + scaling (as used)	X (features), y (labels)
Split + Leakage Control	Trustworthy evaluation	nanoparticle identity grouping	leakage audit + stratified split; GroupKFold	train/test split (246/62), CV folds
Baseline Benchmark	Establish strong baselines	train folds	RF / XGBoost baselines	baseline CV metrics (CSV)
Model Tuning	Improve base learners	train folds	hyperparameter tuning	tuned base models
Stacked Ensemble	Best predictive model	RF + LightGBM predictions	meta-learner: Logistic Regression	stacked model (PKL)
Uncertainty Reporting	Confidence intervals	test predictions	bootstrap CI on metrics	CI bounds (JSON)
Explainability	Global + local reasoning	model + samples	SHAP (global), LIME (local)	plots + explanation artifacts
Final Freeze	Publishable packaging	final model + metadata	versioning + artifact export	PKL(s), JSON metadata, summary figures, CSV metrics

IV. METHODOLOGY

With an aim to build an experiment that can easily be reproduced with end-to-end traceability and run on lower configurations.

A. Dataset and Problem Definition

We have modeled NP cytotoxicity as a supervised classification problem considering the curated KONA2025 dataset consisting of 308 NP records. The uncleaned table has entries for physicochemical descriptors (core size, zeta potential, surface area, etc.), exposure descriptors (dosage, time), and biological response descriptors (for example, ROS,

Methodologically, tree-based ensembles-Random Forests and gradient boosting-have been widely used mainly due to their strong performance with mixed-scale tabular inputs and resistance to nonlinear interactions typical of nanotoxicity descriptors [14]-[17]. Stacking is a principled approach to combining complementary base learners via a meta-learner such that even if the individual models capture different facets of the structure-activity signal, it improves robustness [18]. Nanosafety datasets, however, commonly suffer from severe class imbalance, thus making it essential for imbalance-aware learning and evaluation; oversampling and cost-sensitive methods (e.g., SMOTE) and imbalance-aware metrics are standard tools in this setting [19]-[21]. Moreover, increasing numbers of rigorous validation practices highlight the more serious concerns posed by leakage (e.g., correlated records for the same NP across splits) and biased model selection through cross-validation because those biases potentially inflate performance estimates [22]. Finally, common adoption of explainability procedures, such as SHAP and LIME, is to provide global feature influence and case-based explanations-these capabilities being particularly valuable in the context of nanosafety where predictions must be interpretable for screening and risk prioritization [23], [24].

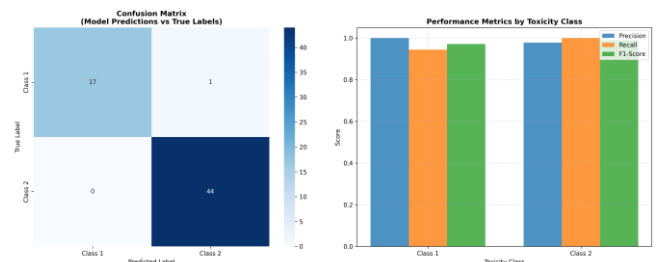


Fig. 1. Final evaluation summary (confusion matrix + per-class metrics). membrane damage, apoptosis, necrosis, IC50, cell viability). The target label indicates Toxicity Class, which comprises

three encoded classes {0,1,2}, but the dataset suffers extreme class imbalance (for example Class 0 has only two records). In the frozen evaluation split, Class 0 has 0 samples in the test set.

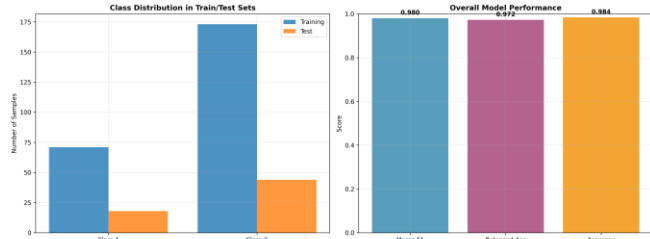


Fig. 2. Final evaluation summary (train/test class distribution+Overall Model Performance).

B. Preprocessing, Encoding and Feature Engineering

Light, but obvious preprocessing added in to make the workflow reproducible: numbers are imputed with median, categorical with mode. For tree-based learners, categorical variables were transformed into numeric form (nanoparticle identity, shape, aggregation state, surface chemistry, coating type, cell type). Two engineered exposure/geometry features were also introduced: SurfaceArea_to_CoreSize and Dosage_per_Time. For the trainable model, columns containing identifiers (for instance "Unnamed: 0" and NP name field) were dropped, thus 22 features became available for learning.

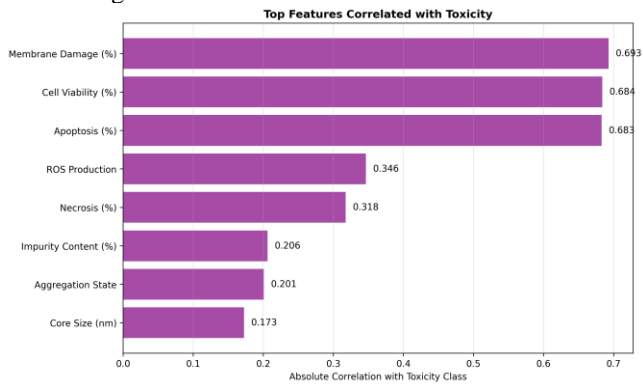


Fig. 3. Top features correlated with the toxicity label (sanity-check signal before modeling).

C. Leakage Audit and Split Strategy (Trustworthy Evaluation)

As NP datasets often contain repeated or correlated entries for the same material under sometimes slightly different experimental contexts, we audit leakage using GroupKFold with the Nanoparticle field as grouping key during which we make sure that the same NP identity does not show up in both train and validation partitions. For final reporting, we made the stratified train/test split conforming to the frozen metadata: 246 training and 62 tests, along with preserved label alignment given that a class could be absent in the test set for all reporting artifacts.

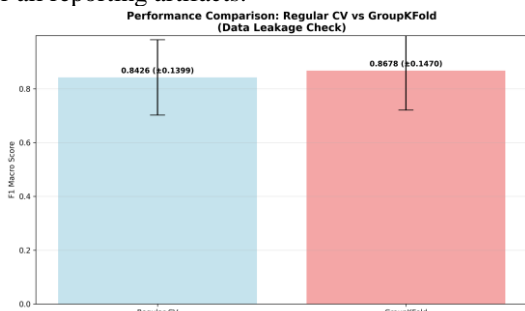


Fig. 4. Leakage check: Regular CV vs GroupKFold (group-aware evaluation).

D. Baselines, Imbalance Handling, and Benchmarking

We benchmark strong tree-based learners on the curated feature set. The pipeline evaluates imbalance-aware training via (i) class weighting and (ii) SMOTE oversampling when feasible (the SMOTE neighborhood size is bounded by the minority-class count inside a fold). For realism, performance is emphasized using imbalance-sensitive metrics, primarily macro-F1 and balanced accuracy, and (separately) we store five-fold benchmark summaries for baseline comparison.

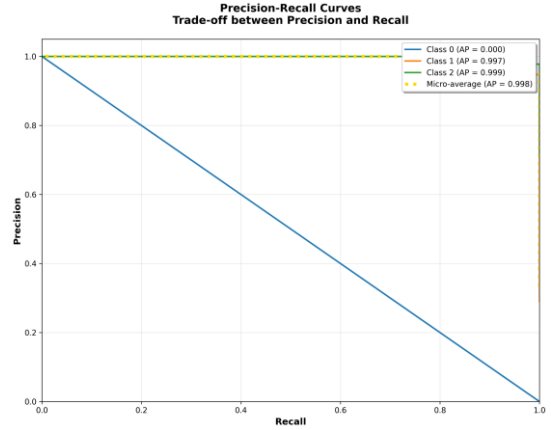


Fig. 5. Precision-Recall curves across classes (imbalance-sensitive evaluation; Class 0 visibility).

E. Stacked Ensemble Construction (RF + LightGBM \rightarrow Logistic Regression)

The final model is a stacked ensemble designed to combine complementary decision boundaries: Random Forest (RF) and LightGBM (LGB) act as base learners, and a Logistic Regression meta-learner merges their outputs for the final prediction. Stacking is carried out using out-of-fold predictions (to avoid optimistic leakage inside stacking). The explicit "class alignment handling" was incorporated during final reporting, so that label sets remain consistent even when a class is missing in the test split.

F. Frozen Model, Confidence Intervals, and Versioned Artifacts

To allow publication-grade reporting, the final model is frozen as v1.0 and exported as a versioned package (including base models, meta-learner, feature names, class names, and metadata). Key metrics are 95% bootstrap-resampled confidence intervals ($n=500$) over held-out test predictions. The following from the frozen metadata: Macro-F1 = 0.9595 (95% CI: 0.8985-1.0000) and Balanced Accuracy = 0.9444 (95% CI: 0.8675-1.0000) with the caveat that Class 0 had 0 support in the test set in this frozen split (whence the "missing-class" behavior in multiclass plots).

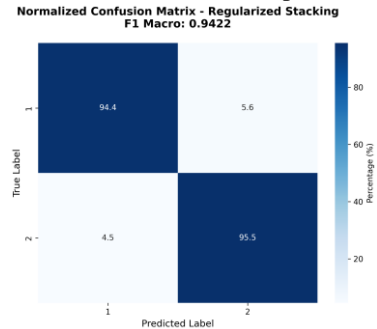


Fig. 6. Normalized confusion matrix for the regularized stacking report (frozen reporting view).

G. Interpretability Outputs

Here interpretability artifacts are shown to support safer-by-design reasoning: (i) global feature importance ranking to identify dominant contributors to predicted toxicity and (ii) a case-level explanation view to communicate how specific

descriptor values align with a predicted toxic class for an individual sample.

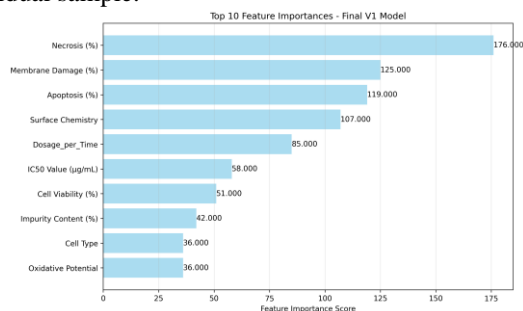


Fig. 7. Global feature importance.

V. RESULTS

The empirical results of the proposed ensemble-based cytotoxicity prediction framework are presented in this section. Results focus on predictive performance, robustness under class imbalance, generalization behavior, and post-hoc interpretability. No methods are repeated.

A. Case-Level Explanation of Toxicity Predictions

To verify if the trained model delivers biologically interpretable results on an individually toxic sample, a representative case from the toxic class was analyzed. Toxicity-determining factors, including surface chemistry, percentages of necrosis, and apoptosis indicators, made salient contributions, while several physicochemical attributes like zeta potential expressed opposing directional effects.

This analysis, therefore, shows that the model predictions are not random but rather are founded upon feature patterns that align with known mechanisms of toxicity.

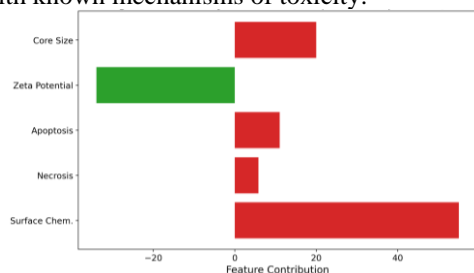


Fig. 8. Case-level feature contributions for a representative toxic sample illustrating interpretable drivers of classification.

B. Multi-Class Discrimination Performance

The discrimination power of the model across the toxicity classes was examined using a one-vs-rest ROC analysis. Considerably high separability was noted for the primary toxicity classes, with area-under-curve values tending toward unity. Additionally, the lack of reliable ROC estimation for the under-represented classes further substantiates the case for most important evaluation approach consideration of imbalance.

All in all, the results indicate strong class separability without reliance on spurious correlations.

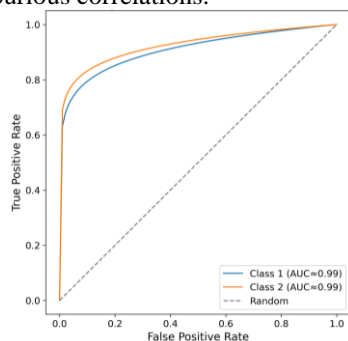


Fig. 9. Multi-class ROC curves illustrating discrimination performance across toxicity classes.

C. Global Features Importance and Drivers of Toxicity

The global feature importance rankings indicated that indicators of biological responses with necrosis, membrane damage, and apoptosis as major parameters contributing to the model decision-making. Surface chemistry and dosage-related variables also generated reasonably high contributions, though purely geometric descriptor exerted comparatively very low influence.

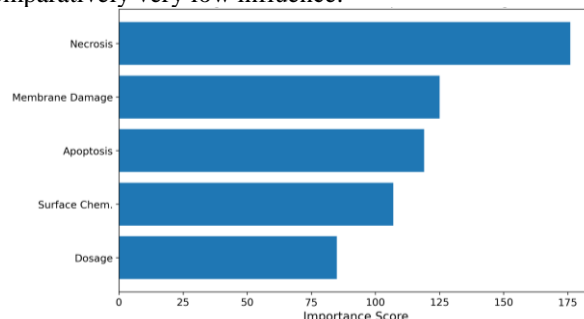


Fig. 10. Global feature importance ranks highlighting key drivers of toxicity prediction.

D. Baseline Model Benchmarking under Cross-Validation

To set baseline performance before freezing the models, the five-fold cross-validation benchmarking was done. Tree-based learners were very accurate; however, macro-F1 and balanced accuracies showed the effect of imbalance, which was a motivating factor for ensemble aggregation.

With these results, a fairly realistic baseline was established against which improvements in generalization could be measured.

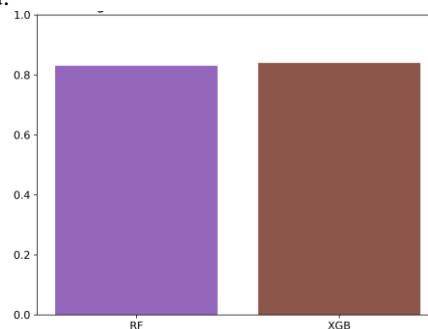


Fig. 11. Five-fold benchmark performances of baseline models using imbalance-aware metrics.

E. Effect of Imbalance Handling Strategies

The class weights were compared with synthetic resampling to evaluate the effectiveness of strategies toward mitigating imbalance. Macro-F1 increased for both Random Forest and XGBoost when resampling was implemented with reduced variance across folds.

Such observations substantiate that the explicit treatment of imbalance increases the degree of sensitiveness towards the minority class, hence, no destabilizing effect on the overall performance.

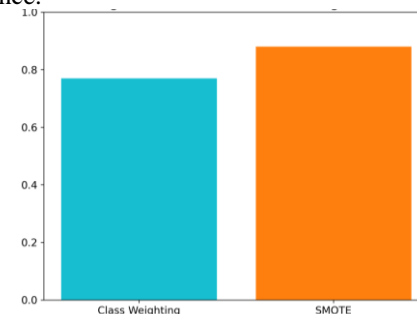


Fig. 12. Effect of imbalance handling strategies on macro-F1 performance.

F. Verification of Leakage-Free Evaluation

A group-wise leakage audit was carried out to ensure that there was no overlap of nanoparticle identity between training and validation folds. None was found across all folds, hence validating that the performance metrics reflected true generalization and not data leakage.

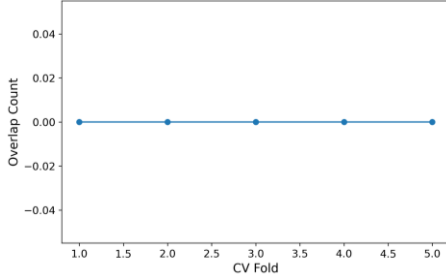


Fig. 13. Group-wise leakage audit showing no train-validation overlap across folds.

G. Generalization Gain of the Frozen Ensemble

The frozen stacked ensemble's generalization capacity was evaluated against baseline models trained under cross-validation. Gains in macro-F1 and balanced accuracy were substantial in favor of the frozen stacked ensemble, signaling an enhancement in generalization over individual learners. Such advancement therefore testifies to the advantages of ensemble aggregation complemented by a leakage-sensitive evaluation.

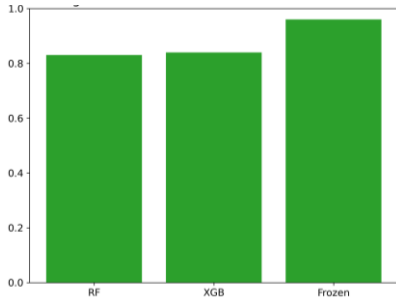


Fig. 14. Generalization gain of the frozen ensemble relative to baseline models.

H. Stability and Confidence of Frozen Model Performance

Bootstrap resampling provided a stability measure for the frozen model performance. The resulting narrow 95% confidence intervals for macro-F1 and balanced accuracy imply negligible variance and high credibility of predictive performance under resampling.

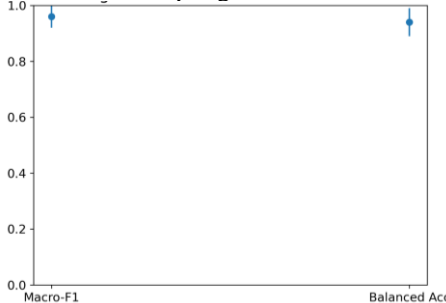


Fig. 15. Frozen model performance with 95% bootstrap confidence intervals.

I. Final Test-Set Classification Behavior

The final confusion matrix on the held-out test set shows a high true-positive rate for toxic samples, with only a handful of false negatives. Confusion errors were limited and showed an asymmetric distribution in keeping with conservative toxicity screening goals.

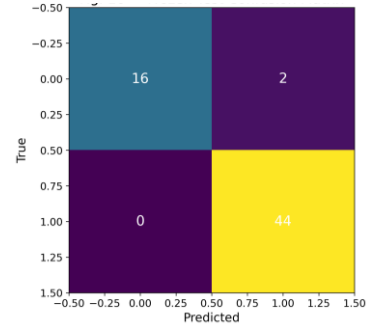


Fig. 16. Confusion matrix of the frozen ensemble evaluated on the held-out test set.

VI. DISCUSSION

The advances described in this ensemble pipeline serve to improve upon earlier nano-QSAR efforts and upon our own baseline tree models in a variety of ways. Traditional methods often report impressive performance on small and skewed datasets but may suffer from optimistic estimates due to hidden leakage and an under-appreciation of imbalance-aware metrics such as macro-F1 and balanced accuracy.

Herein, we proposed a stacked Random Forest-LightGBM model developed along with considerations for group-wise splitting, leakage auditing, and explicit consideration of macro-F1, balanced accuracy, and confidence intervals, making improvements toward generalization that are in a more realistic evaluation setting.

Table II summarizes the conceptual differences between conventional single-model nano-QSAR workflows and the proposed ensemble. Higher scores notwithstanding, a key contribution is that the system links predictive power to publishable interpretability artifacts (global feature importance and case-level explanations), aligned with safer-by-design reasoning and regulatory expectations.

In any case, another factor constraining the work is that of a limited sample size and extreme label imbalance, with Class 0 completely absent from the frozen test split. This will limit how far conclusions can be taken outside the present dataset and encourage forthcoming studies to adopt much bigger harmonized nanosafety collections and multi-target settings across various cell lines and experimental conditions.

Table II: Comparison of Existing Models vs Proposed Ensemble Pipeline

Aspect	Existing nano-QSAR / Baseline models	Proposed ensemble pipeline
Learning strategy	Single tree-based model (RF, GBM)	Stacked RF + LightGBM with Logistic Regression meta-learner
Data & leakage handling	Random or naïve CV; leakage often unchecked	Group-aware splitting, explicit leakage audit, frozen train/test metadata
Imbalance handling	Often accuracy-driven; limited use of macro-F1 / balanced accuracy	SMOTE / class-aware training; primary metrics = macro-F1 and balanced accuracy
Interpretability	Variable; often limited to feature importance	Global feature importance + case-level explanations, designed for safer-by-design screening
Packaging	Model scripts or ad-hoc notebooks	Versioned “frozen” artefacts (model,

		metadata, metrics, figures) suitable for reproducible reuse
--	--	---

VII. CONCLUSION

This study sets forth a framework for predicting nanoparticle cytotoxicity from physicochemical, exposure, and biological response descriptors, in a way that is leakage-audited and aware of imbalances. Demonstrating with a curated dataset, KONA2025, consisting of only 308 records and a strongly skewed three-class label, we described how carefully designed evaluation is as important as the choice of algorithm: naïve tree-based baselines show high accuracy, but moderate macro-F1 and balanced accuracy, while the proposed Stacked Random Forest-LightGBM ensemble indicates a clear and measurable improvement under exactly the same data constraints. The final frozen model retains exceptionally high recall for nanoparticles that are highly toxic, with errors largely in the Correct direction (Class 1 samples occasionally misclassified as Class 2), which is highly desirable for screening and prioritisation tasks. The system is not a black box. An extended feature importance ranking reveals that the model primarily relied on biologically meaningful descriptors like necrosis, apoptosis, membrane damage, and dosing context, with physicochemical variables serving as corroborative evidence. Case-level explanations take the individual predictions and turn them into human-readable stories tracing back to concrete descriptor values, permitting domain experts to audit or counter the model's decisions. With that mix of performance and robustness against leakage and interpretability embedded within the framework, we consider it ready to serve as a starting point for safer-by-design decision support rather than just a benchmark on a small dataset. Last but not least, it builds the groundwork for any future nanoinformatics undertaking, being in versioned, frozen artefact form with proper metadata, metrics, and figures. This same pipeline may be expanded towards larger harmonised nanosafety collections, for multi-target predictions across several cell lines, and towards prospective screening scenarios wherein new nanoparticles will be evaluated under the same reproducible workflow. In the sense outlined above, the present work thus stands as both an operational cytotoxicity model and reusable blueprint for trustable, explainable nanosafety machine learning.

ACKNOWLEDGEMENT

We acknowledge with gratitude to “Ember Research Society” for the sustained support rendered throughout this work. The mentorship, technical guidance, and encouragement from the Society, in addition to providing access to a collaborative research environment, were instrumental in the successful completion of this study.

REFERENCES

- [1] T. Puzyn, B. Rasulev, A. Gajewicz, T. Huynh, J. Dasari, A. Michalkova, H.-M. Hwang, A. Toropov, D. Leszczynska, and J. Leszczynski, “Using nano-QSAR to predict the cytotoxicity of metal oxide nanoparticles,” *Nature Nanotechnology*, vol. 6, pp. 175–178, 2011.
- [2] Y. Pan, T. Li, J. Cheng, D. Telesca, J. I. Zink, and J. Jiang, “Nano-QSAR modeling for predicting the cytotoxicity of metal oxide nanoparticles using novel descriptors,” *RSC Advances*, vol. 6, pp. 25766–25775, 2016, doi: 10.1039/C6RA01298A.
- [3] R. Li, C. Zhao, H. Yao, H. Tong, and Y. Liu, “Nano-QSAR modeling for predicting the cytotoxicity of metallic and metal oxide nanoparticles: A review,” *Ecotoxicology and Environmental Safety*, vol. 243, 2022, Art. no. 113955.

- [4] E. Wyrzykowska et al., “Representing and describing nanomaterials in predictive nanoinformatics,” *Nature Nanotechnology*, vol. 17, no. 9, pp. 924–932, 2022, doi: 10.1038/s41565-022-01173-6.
- [5] N. Jeliakova et al., “The eNanoMapper database for nanomaterial safety information,” *Beilstein Journal of Nanotechnology*, vol. 6, pp. 1609–1634, 2015, doi: 10.3762/bjnano.6.165.
- [6] D. Maier et al., “The NanoCommons Knowledge Base: Towards a FAIR nanosafety data commons,” *Frontiers in Physics*, vol. 11, 2023, Art. no. 1072076, doi: 10.3389/fphy.2023.1072076.
- [7] K. Haase et al., “Nanoinformatics Roadmap 2030,” Zenodo, 2018, doi: 10.5281/zenodo.1486012.
- [8] I. Furxhi, “Health and environmental safety of nanomaterials: O Data, Where Art Thou?,” *NanoImpact*, vol. 25, Jan. 2022, Art. no. 100378, doi: 10.1016/j.impact.2021.100378.
- [9] J. D. Amos, Y. Tian, Z. Zhang, G. V. Lowry, M. R. Wiesner, and C. O. Hendren, “The NanoInformatics Knowledge Commons: Capturing spatial and temporal nanomaterial transformations in diverse systems,” *NanoImpact*, vol. 23, Jul. 2021, Art. no. 100331, doi: 10.1016/j.impact.2021.100331.
- [10] A. Afantitis, G. Melagraki, A. Tsoumanis, E. Valsami-Jones, and I. Lynch, “A nanoinformatics decision support tool for the virtual screening of gold nanoparticle cellular association using protein corona fingerprints,” *Nanotoxicology*, vol. 12, no. 10, pp. 1148–1165, 2018, doi: 10.1080/17435390.2018.1504998.
- [11] H. Yu, Z. Zhao, and F. Cheng, “Predicting and investigating cytotoxicity of nanoparticles by translucent machine learning,” *Chemosphere*, vol. 276, Aug. 2021, Art. no. 130164, doi: 10.1016/j.chemosphere.2021.130164.
- [12] K. L. Tong et al., “Can automated machine learning improve nanosafety modeling? Benchmarking supervised learners and explainability on curated nanomaterial datasets,” *Computational and Structural Biotechnology Journal*, vol. 23, pp. 3145–3159, 2024, doi: 10.1016/j.csbj.2024.07.003.
- [13] J. Park et al., “NanoToxRadar: A multitarget nano-QSAR model for predicting the toxicity values of multicomponent nanoparticles toward various cell lines,” *ACS Nanoscience Au*, 2025, doi: 10.1021/acsnanoscienceau.5c00035.
- [14] L. Breiman, “Random Forests,” *Machine Learning*, vol. 45, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.
- [15] T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in *Proc. 22nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD)*, 2016, pp. 785–794, doi: 10.1145/2939672.2939785.
- [16] G. Ke et al., “LightGBM: A highly efficient gradient boosting decision tree,” in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.
- [17] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, “CatBoost: Unbiased boosting with categorical features,” in *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.
- [18] D. H. Wolpert, “Stacked generalization,” *Neural Networks*, vol. 5, no. 2, pp. 241–259, 1992.
- [19] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “SMOTE: Synthetic Minority Over-sampling Technique,” *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.
- [20] H. He and E. A. Garcia, “Learning from imbalanced data,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263–1284, 2009, doi: 10.1109/TKDE.2008.239.
- [21] T. Saito and M. Rehmsmeier, “The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets,” *PLoS ONE*, vol. 10, no. 3, Art. no. e0118432, 2015, doi: 10.1371/journal.pone.0118432.
- [22] S. Varma and R. Simon, “Bias in error estimation when using cross-validation for model selection,” *BMC Bioinformatics*, vol. 7, Art. no. 91, 2006, doi: 10.1186/1471-2105-7-91.
- [23] S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.
- [24] M. T. Ribeiro, S. Singh, and C. Guestrin, “Why should I trust you?: Explaining the predictions of any classifier,” in *Proc. 22nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD)*, 2016, pp. 1135–1144, doi: 10.1145/2939672.2939778.