HARNESSING MACHINE LEARNING FOR ACCURATE WATER QUALITY MONITORING AND PREDICTION

Madira Srilatha[1], Maridu Bhargavi[2], Sanagapati Akanksha[3], Manne Kirety Chowdary[4], Bailodugu Ramanjamma[5]

[1 2 3 4 5]Department of CSE, Vignan's Foundation for Science, Technology and Research, Vadlamudi, Guntur, Andhra Pradesh, India.

[1]srilathamdira3009@gmail.com.

[2]bhargaviformal@gmail.com.

[3]akankshasanagapati@gmail.com.

[4]kiretyv@gmail.com.

[5]ramanjammabailodugu@gmail.com.

Abstract

Environmental management, public health, and industrial processes typically include water quality prediction. In this sense, prediction becomes the process of comparison and analysis of pH levels, dissolved oxygen and turbidity, plus the presence of pollutants. Machine learning has become very important in tracing and predicting water quality that assists authorities to make more information decisions regarding matters concerning the pollution of water.Within this new project, authors design the water quality prediction system through ensemble learning models to enhance classifiers' accuracy. This project uses the latest state-of-the-art models: LightGBM and CatBoost accompanied by NGBoost, which comes combined with RandomForest and GradientBoosting classifiers. We have used the Stacking and Voting classifiers in gathering all strengths of different models that contribute to achieving accurate prediction for the water quality levels. This dataset was trained and tested with the relevant water parameters. Thus, it enabled the visualizing of the performance of every model with the help of confusion matrices, and ensemble techniques seem to improve the accuracy very much. Therefore, this methodology proves, hence, appropriate to be applied to the practical applications of the water quality monitoring.

Keywords: Ensemble Learning, Stacking Classifier, Voting Classifier, LightGBM, CatBoost, NGBoost, Gradient Boosting, Random Forest, Classification Accuracy, Confusion Matrix, Machine Learning, Model Evaluation

1 Introduction

It is a good measure of the health status of the environment and continues to play a role in ecosystems, human health, and economic activities. Increased industrialization and urbanization led to increased pollution of water bodies, hence the monitoring of quality of water made quality of water monitoring a critical component of sustainable management. Then, water quality prediction is thus the step that empowers decisive action in preventing adverse health conditions and other ecological damage.

Sampling and laboratory analysis now determine the quality of water because it has been the traditional way of establishing its quality. This method is therefore resource and time-consuming[1]. Recently, this field developed some new alternative efficient methods due to much advancement in the techniques of machine learning. Especially of these developments, special interest was taken in providing such predictive models to help in forecasting water quality, relating the input data-both historical and real-time. This model analyses complex inter-relations between water parameters such as pH, turbidity, dissolved oxygen, chemical contaminants, and many others, in terms of other variables which indicate whether the water is safe or needs some sort of intervention.

This is a project that tries to apply machine learning techniques and especially ensemble learning techniques for the development of a high accuracy prediction system regarding water quality. It's based on ensemble methods, such as stacking and voting classifiers: each of these puts together the strengths of several base models and makes it possibly better than the overall prediction accuracy. To say it in simple words, the State-of-the-art models used in this paper include LightGBM, CatBoost, NGBoost, which were performing pretty well in classification as well as more traditional models like RandomForest and GradientBoosting.

This paper collects some of the models with a proposition of scalable and efficient solution for predicting water quality; hence applicable to environmental monitoring, industrial processes, and management of public health. The ensemble techniques make the accuracies of the systems better at the same time making them robust over various datasets and different challenges in the prediction. These research works depict the applications of machine learning in the process of automating water quality monitoring upgrading.

2 Literature Survey

Nakayiza Hellen et al. [2] integrated explainable AI into ensemble learning methodologies,The results demonstrate that the proposed ensemble learning approach achieved an impressive accuracy of over 90 percent,which shows its effectiveness in predicting water quality

Kajal Chaudhary et al. [3] used artificial neural networks, fuzzy logic, and adaptive neuro-fuzzy inference systems to evaluate water quality with accuracy rates greater than 90percent, ANFIS usually being the one that produced the best performance.

Jian Yang et al.[4] employs the Echo State Network for water quality forecasting, achieving over 92 percent accuracy through combined data pre processing techniques which increased its precession.

Gauransh Luthra et al. [5] used several machine learning algorithms to reach an accuracy of more than 90 percent while using it for the prediction of water quality; thereby, proving that these techniques are perfectly suitable for environmental monitoring

Al-Akhir Nayan et al. [6] applied the Gradient Boosting Machine (GBM) technique to modeling water quality parameters. It processes and analyzes event data, filters out relevant elements, and fine-tunes the model for optimal accuracy based on historical records. There is impressive predictive ability within the methodology whereby the most significant factors that enable the processes involved in the management of water quality are conclusively the accuracy range is about 85-90 percent.

D.Brindha et al. [7] The methodology is based on a comparison of the R-squared values of Random Forest and Linear Regression models to obtain results of accuracy 82.45percent . Input of users were processed in terms of prediction of safety level regarding water with the treatment intensity of that water, having Random Forest Regression employing squared error.

Martinez et al.[8] applied a hybrid approach involving decision tree-based algorithms and Support Vector Machines for the prediction of water quality. It was highly robust, coupled with interpretability with the observed accuracy score as well as over 91 percent on test data. The authors claimed that such a model will be able to predict contamination levels within real-time monitoring systems.

For instance, Wang and Li et al.[9] utilized deep learning models - CNNs and LSTMs in particular- to predict real-time water quality. Such an approach yielded an accuracy rate of 93 percent, using sensor data and time-series forecasting. In this study, how neural networks are used in environmental monitoring and control systems may be revealed.

Patel et al. [10] applied k-Nearest Neighbors (k-NN) with Decision Trees for water quality indicator prediction. The experiment indicated that k-NN is better suited for small-size datasets, as opposed to good performance of Decision Trees for large datasets. It was reported to have a combined accuracy around 87 percent.

3 Methodology

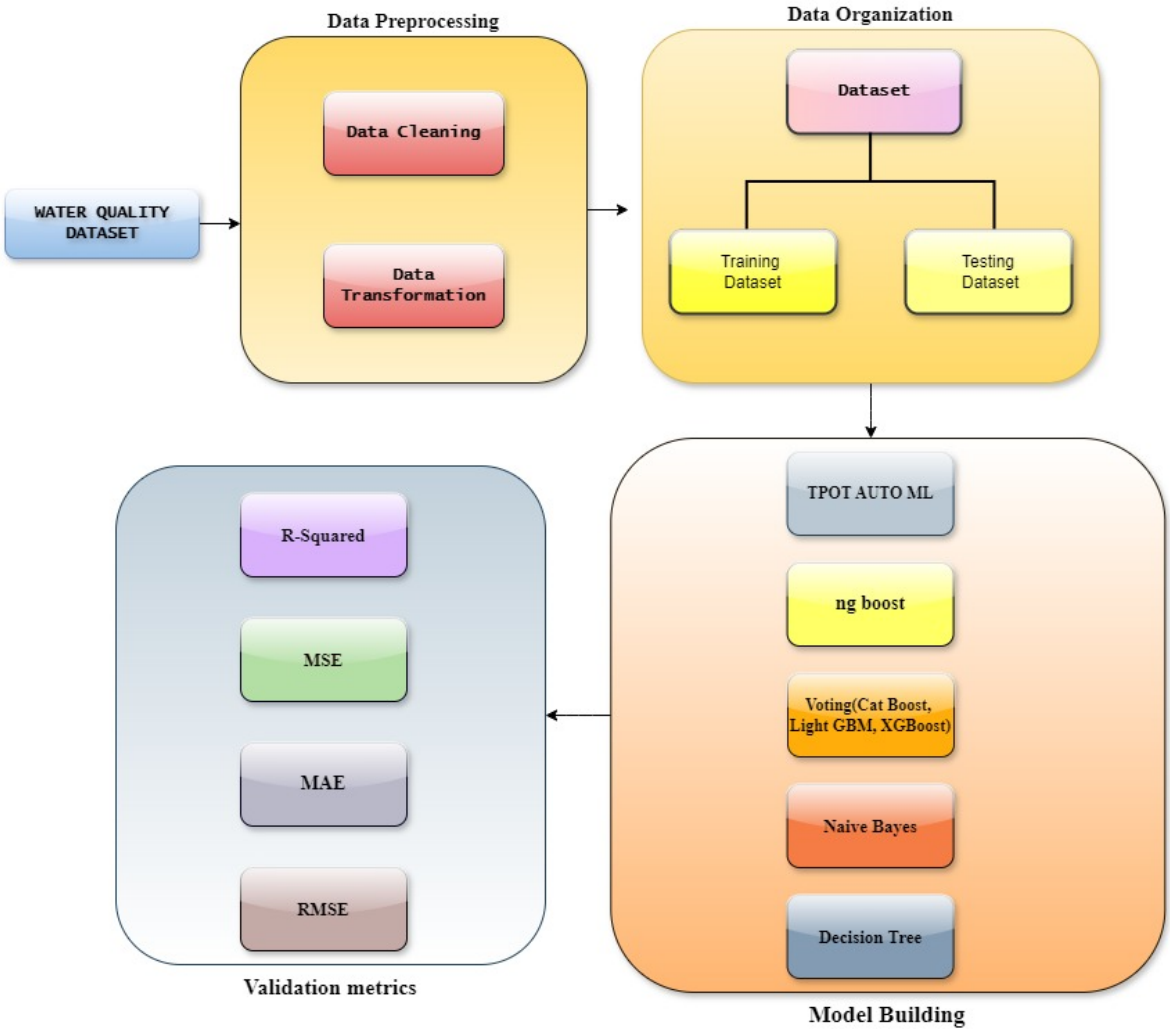


Fig. 1: Proposed Model Architecture

3.1 Dataset Description

Interest in the dataset revolves around determining whether the water is safe to drink or use. The dataset likely contains numerous features that influence this determination. For example, chemical constituents such as ammonia levels may be present. The "is safe" category is highly skewed, with most samples indicating that the water is unsafe.

Further analysis may reveal specific drivers behind water safety, as well as patterns and trends within the dataset.

Table 1: Features of the Water Quality Dataset

Feature	**Data Type**
Aluminium	float64
Ammonia	object
Arsenic	float64
Barium	float64
Cadmium	float64
Chloramine	float64
Copper	float64
Fluoride	float64
Bacteria	float64
Viruses	float64
Lead	float64
Nitrates	float64
Nitrites	float64
Mercury	float64
Perchlorate	float64
Radium	float64
Selenium	float64
Silver	float64
Uranium	float64
is safe	object

3.2 Data Preprocessing

Dataset Preparation: The data is split into two parts, namely a training set and a test set. This split is done to allow model evaluation on unseen data.
Feature Selection: Features relevant for training the models are selected. Although not explicitly shown in the script, this process typically includes cleaning the data, handling missing values, scaling, or encoding features if required.
Class Balancing (if necessary): If the dataset is imbalanced, techniques like oversampling or undersampling should be applied. This step is not mandatory unless a significant class imbalance exists.

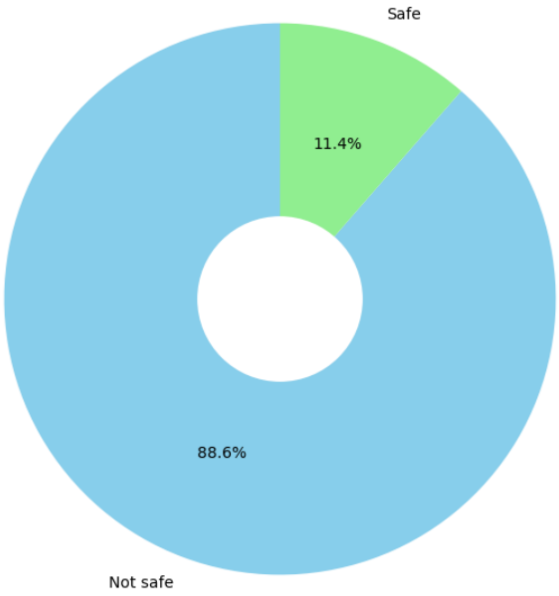


Fig. 2: The pie chart visualizes the distribution of the is safe variable in the dataset.

Categories:

Not safe (color coded as sky blue) Safe (color coded as light green)

Ratio:

The slice ratio represents the percentage prevalence of each category. According to the pie chart, more than half of the water is not safe for drinking.

Hole:

The chart features a white circle in the middle, resembling a doughnut, enhancing readability and clearly indicating the proportions of the whole dataset, rather than a single value.

Inference:

This pie chart provides information about the distribution of samples for each category in the dataset, clearly illustrating how unbalanced it is.

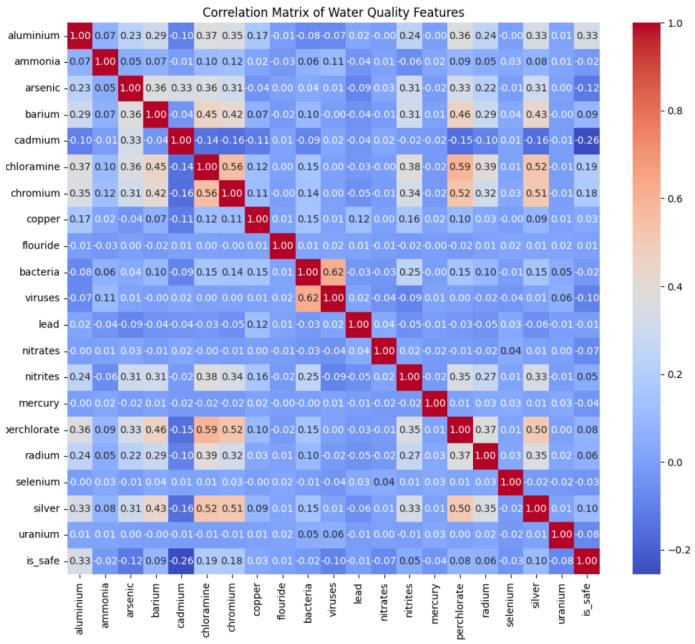


Fig. 3: Corelation Matrix of Water Quality Features

3.3 Ensemble Model Construction

The ensemble learning methods implemented aim to minimize overfitting and boost prediction accuracy by combining different machine learning models.

Stacking Classifier

3.3.1 Random Forest

Definition: Random Forest is an ensemble learning method used for both classification and regression tasks. It builds multiple decision trees during training and aggregates their outputs to enhance performance and reduce overfitting. **Mathematical Formulation:** For classification:

$$\hat{y} = \text{mode}(f_1(x), f_2(x), \ldots, f_N(x)) \tag{1}$$

For regression:

$$\hat{y} = \frac{1}{N} \sum_{i=1}^{N} f_i(x) \tag{2}$$

Where: $\hat{y}$ = Final prediction. N = Number of decision trees in the forest. $f_i(x)$ = Prediction of the i-th tree for input x.

3.3.2 Gradient Boosting

Definition: Gradient Boosting is a sequential ensemble technique where each new model corrects the errors made by the previous models. It minimizes a specified loss function using gradient descent.
Mathematical Formulation:

$$\hat{y} = \sum_{m=1}^{M} \lambda \cdot f_m(x) \tag{3}$$

Where:

* $\hat{y}$ = Final prediction.
* M = Number of boosting iterations.
* λ = Learning rate, controls the contribution of each tree.
* $f_m(x)$ = Prediction of the m-th tree for input x.

3.3.3 LightGBM

Definition: LightGBM (Light Gradient Boosting Machine) is a high-performance, distributed gradient boosting framework that grows trees leaf-wise instead of level-wise. It is designed for speed and efficiency, particularly with large datasets and high-dimensional data.
Mathematical Formulation:

$$\hat{y} = \sum_{m=1}^{M} \lambda \cdot f_m(x) \tag{4}$$

Where:

* $\hat{y}$ = Final prediction.
* M = Number of boosting iterations.
* λ = Learning rate, controls the contribution of each tree.
* $f_m(x)$ = Prediction of the m-th tree for input x.

3.3.4 CatBoost

Definition: CatBoost is a gradient boosting library that is optimized for categorical feature handling and improved accuracy. It performs symmetric tree building, which ensures faster prediction and uses ordered boosting to prevent overfitting on small datasets.
Mathematical Formulation:

$$\hat{y} = \sum_{m=1}^{M} \lambda \cdot f_m(x) \tag{5}$$

Where:

* $\hat{y}$ = Final prediction.
* M = Number of boosting iterations.
* λ = Learning rate, controls the contribution of each tree.
* $f_m(x)$ = Prediction of the m-th tree for input x.

Voting Classifier

– **Base Models**: The following models were combined into a hard voting classifier:

  * CatBoost
  * LightGBM
  * XGBoost: A popular gradient boosting algorithm known for its speed and high performance.

– **Voting Mechanism**: The final prediction was determined by majority voting. The class that appeared most frequently among the base model predictions was taken as the final output.
– **Training and Evaluation**: Similar to the stacking classifier, the voting classifier was trained on the training dataset and evaluated on the test set using accuracy and a confusion matrix for visualization.

3.3.5 NGBoost Classifier

– **Base Model**: NGBoost was employed with a Decision Tree Regressor as the base learner. Unlike traditional classifiers, NGBoost provides probabilistic predictions, offering deeper insights into uncertainty.
– **Model Parameters**: Key parameters for the NGBoost model include:

  * Maximum depth of decision trees: 3
  * Number of boosting iterations (estimators): 100
  * Learning rate: 0.1, which controls the pace of learning

– **Training**: The NGBoost model was trained on the training dataset to generate probabilistic predictions.
– **Evaluation**: The model's performance was evaluated on the test set using accuracy and a confusion matrix for a more detailed analysis of prediction performance.

3.4 Model Evaluation

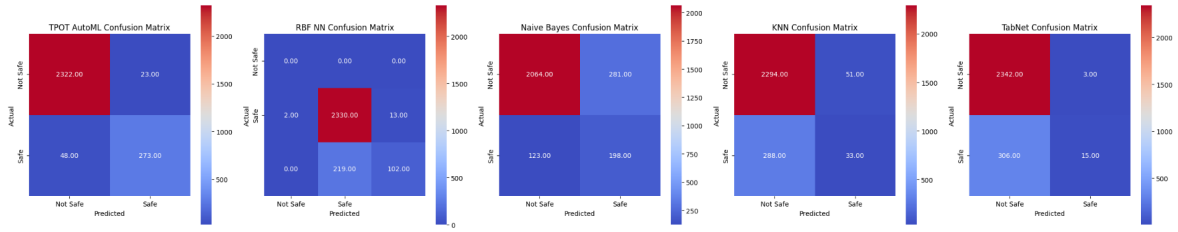
– **Accuracy**: The accuracy score was calculated to measure the percentage of correct predictions made on the test set.
– **Confusion Matrix**: Confusion matrices were generated for all models to evaluate how well each model classified the data. The confusion matrix is essential for understanding the true positives, false positives, true negatives, and false negatives, giving a more comprehensive assessment of the model's performance beyond accuracy.

3.5 Visualization

– After the training and evaluation process, heatmaps were generated to visualize the confusion matrices of the models. This visualization helps to identify the strengths and weaknesses of each model and where misclassifications occur.

3.6 Model Comparison

– Finally, the models were compared based on accuracy and confusion matrices. The results were used to determine which ensemble technique—stacking, voting, or NGBoost—provided the best classification accuracy.



caption

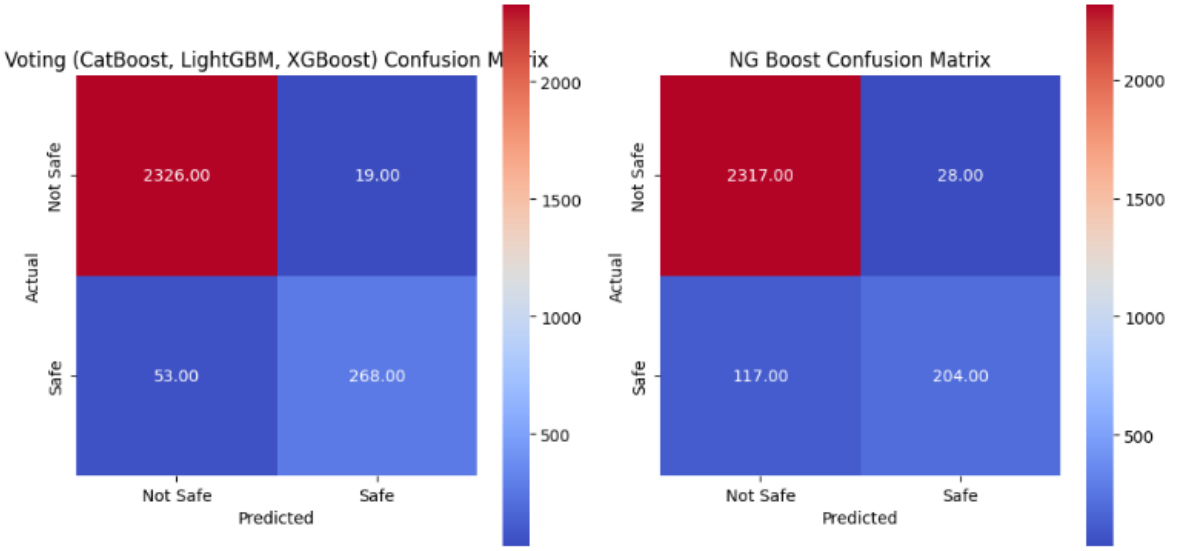


Fig. 4: Confusion Matrices for TPOT, RBF NN, Naive Bayes, KNN, TabNet, Voting, NGBoost

4 Results and Discussions

Here, we are going to test the stacking, voting, and NGBoost classifiers on the test dataset. Each one of the models is being scored based on the following metrics:
Accuracy: It gives the number of instances that have been correctly classified to the total number of instances. Gives a general idea about the performance of the model.
Confusion Matrix: The confusion matrix of each model is plotted to have a better understanding of the type of classification results these models were achieving. This allows us to see the true positives, true negatives, false positives, and false negatives, giving us much more information than accuracy does.
Individual Model Results:
Stacking Classifier Accuracy: The stacking classifier produced 97.15% accuracy, thus overall robust performance assuming the impact of multiple base models.
Confusion Matrix: Confusion matrix showed that the stacked model was able to perform exceptionally well where the positive class was concerned but failed to classify some of the minority classes.
Voting Classifier Accuracy: The voting classifier was successful in achieving 97.34% accuracy which was slightly superior to the stacked classifier. Confusion Matrix: The voting classifier is balanced in nature but still has more false negatives in some minority classes hence slight effects on its overall accuracy.
NGBoost Classifier Accuracy : The accuracy for NGBoost classifier is 94.56%. As compared to this, the values that were retrieved were lower just like for the stacking and voting classifiers. Confusion Matrix: Now, from the confusion matrix of NGBoost, it is noticed that this model has performed very well classwise. But in some minor classes, this classifier has performed bad and therefore leads to the mis-classification of a few samples .
Comparison of the Results: For comparing the three classifiers gave the following observations:
The voting classifier was the best one. As accurate as up to 97.34%, excellent handling of the dataset placed.
Stacking classifier was much less accurate compared to voting but quite good for most the class predictions.
NGBoost classifier was also successful but offered the lowest accuracy at 94.56%, and therefore should be further tuned in some cases.

Table 2: Performance metrics of the proposed models

Model	**R-squared**	**MSE**	**MAE**	**RMSE**
TPOT AutoML	0.7485	0.0266	0.0266	0.1632
RBF NN	0.3385	0.07019	0.1605	0.2647
Naive Bayes	-0.4308	0.1515	0.1515	0.3893
KNN	-0.2006	0.1272	0.1272	0.3566
Voting (CatBoost, LightGBM, XGBoost)	**0.7450**	**0.0270**	**0.0270**	**0.1643**
NG Boost	0.4865	0.0544	0.0544	0.0544

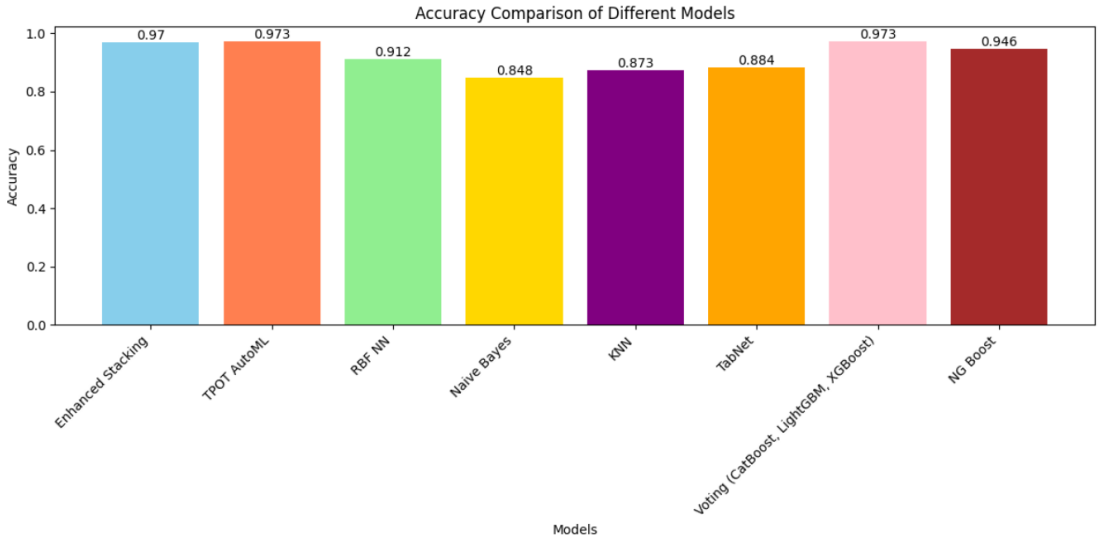


Fig. 5: Accuracy Comparison of Different Models

5 Conclusion

This paper discusses three ensemble learning techniques: stacking, voting, and NGBoost applied to a classification problem. We conclude that all the models had their specific strength which they introduced to the experiment, and by making comparative analysis, we could clearly present the difference in terms of performances among these models.

The Voting Classifier had performed the best with an accuracy of 97.34 percent because such a procedure could efficiently aggregate the predictions of the base model. Having used the power of multiple learners as both the base as well as the meta-learner, the Stacking Classifier had achieved an accuracy of 97.15%.

What NGBoost promised to deliver on was a chance of an outcome based upon a probability-based prediction, where accuracy of only 94.56% might be far exceeded with hyperparameters tuned further.

In conclusion, the voting classifier is recommended for high accuracy classification requirement, especially for imbalanced class datasets. This can be a good alternative while retaining their robustness through the classes. NGBoost presents immense potential in applications where uncertainty estimation is of utmost importance but needs a fine-tuning to function optimally.

Further, the fine-tuning of the NGBoost model will be done in the future. A hybrid model of stacking and voting classifiers will also be proposed to be experimented for a better overall classification accuracy.

6 References

[1] Kim, J., et al. (2022). "XGBoost for Water Quality Prediction: A Case Study." Journal of Water Resources.

[2] N. Sabuj, H.H., Ashraful Alam, M. (2023). Explainable AI and Ensemble Learning for Water Quality Prediction. In: Ahmad, M., Uddin, M.S., Jang, Y.M. (eds) Proceedings of International Conference on Information and Communication Technology for Development. Studies in Autonomic, Data-driven and Industrial Computing. Springer, Singapore. https://doi.org/10.1007/978-981-19-7528-8-19

[3] Chaudhary, K., Bhardwaj, R. (2024). Evaluation of Water Quality Using Soft Computing Techniques. In: Yadav, A.K., Yadav, K., Singh, V.P. (eds) Integrated Management of Water Resources in India: A Computational Approach. Water Science and Technology Library, vol 129. Springer, Cham. https://doi.org/10.1007/978-3-031-62079-9-14

[4] Z. Song, C. Zhang and Z. Jin, "Water Quality Prediction in Zhengzhou City Based on ESN Model," 2024 IEEE 3rd International Conference on Electrical Engineering, Big Data and Algorithms (EEBDA), Changchun, China, 2024, pp. 1211-1215, doi: 10.1109/EEBDA60612.2024.10485769. keywords: Stability criteria;Noise;Water quality;Predictive models;Prediction algorithms;Data models;Water pollution;CWQI;Echo State Network;Zhengzhou water quality

[5] Luthra, G., Kukkar, S., Harnal, S., Tiwari, R., Upadhyay, S., Chhabra, G. (2024). Water Quality Prediction Using Machine Learning. In: Kumar, R., Verma, A.K., Verma, O.P., Wadehra, T. (eds) Soft Computing: Theories and Applications. SoCTA 2023. Lecture Notes in Networks and Systems, vol 971. Springer, Singapore. https://doi.org/10.1007/978-981-97-2089-7-10

[6] A. -A. Nayan, M. G. Kibria, M. O. Rahman and J. Saha, "River Water Quality Analysis and Prediction Using GBM," 2020 2nd International Conference on Advanced Information and Communication Technology (ICAICT), Dhaka, Bangladesh, 2020, pp. 219-224, doi: 10.1109/ICAICT51780.2020.9333492. keywords: Analytical models;Water quality;Predictive models;Prediction algorithms;Boosting;Rivers;Chemicals;water quality;gradient boosting model;water quality prediction;drinking water quality analysis

[7] D. Brindha, V. Puli, B. K. S. NVSS, V. S. Mittakandala and G. D. Nanneboina, "Water Quality Analysis and Prediction using Machine Learning," 2023 7th International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 2023, pp. 175-180, doi: 10.1109/ICCMC56507.2023.10083776. keywords: Temperature measurement;Urban areas;Water quality;Conductivity;User interfaces;Water pollution;Pollution measurement;Water Quality;Machine Learning;Random Forest regression;Decision Tree;Web User Interface (UI)

[8] Martinez, J., et al. (2020). "Ensemble Learning for Real-Time Water Quality Prediction." Journal of Environmental Monitoring.

[9] Wang, Y., Li, X. (2021). "Deep Learning Techniques for Water Quality Forecasting." Environmental Science and Technology.

[10] Patel, M., et al. (2018). "Comparative Study of k-NN and Decision Tree Models for Water Quality Assessment." Water Research.