IT Journal Research and Development (ITJRD)
Vol.10, No.2, March 2026, E-ISSN : 2528-4053 | P-ISSN : 2528-4061
DOI : 10.25299/itjrd.2025.22391

Comparative Evaluation of Machine Learning Models For Municipal Solid Waste Prediction With Feature Extension

Mwila Milandile, Muwanei Sinyinda
Departement of Computer Science and Information Technology, Mulungushi University, Zambia
Milandile.mwila1@gmail.com, msinyinda@mu.ac.zm

Article Info

Article history:

Received May 11, 2025
Revised May 21, 2025
Accepted Sep 18, 2025

Keyword:

Machine Learning
Feature Extension
Municipal Solid Waste
Ensemble Learning

ABSTRACT

This paper explores the utilization of machine learning approaches to accurately predict municipal solid waste generation based on two distinct prediction methods: a single-model approach and a multi-model ensemble approach, incorporating feature extension. The predictive performance of these two methods is compared using the metrics Mean Absolute Percentage Error (MAPE), Root Mean Square Error (RMSE), and Mean Absolute Error (MAE). The findings indicate that the multi-model ensemble approach outperformed the single-model method by achieving lower MAPE, RMSE, and MAE values. Specifically, the ensemble model obtained a MAPE of 37.38%, an RMSE of 7,610.76, and a MAE of 5,760.89, while the single-model technique achieved a MAPE of 42.58%, an RMSE of 8,258.01, and a MAE of 6,470.14. These results suggest that combining multiple models can create a more robust and accurate prediction system. Overall, the findings presented in this paper suggest that integrating feature extension and utilizing multiple models results in more accurate predictions, leading to more effective waste management practices.

Corresponding Author:

Mwila Milandile
Department of Computer Science and Information Technology,
Mulungushi University
Kabwe, Zambia
Email : Milandile.mwila1@gmail.com

1. INTRODUCTION

Managing municipal solid waste (MSW) is a critical component of urban sustainability and environmental protection. As urban populations grow and consumption patterns evolve, waste generation has become increasingly complex, posing significant challenges to effective waste management. Sustainable waste management practices are essential not only for protecting the environment but also for safeguarding public health and maintaining the quality of urban living.

Traditionally, MSW management relied on fixed collection schedules and conventional methods. However, the dynamic nature of urban areas characterized by fluctuating population densities, socioeconomic diversity, and changing consumption behaviors demands a more adaptive and data-driven approach. Accurate forecasting of waste generation plays a pivotal role in optimizing collection routes, reducing operational costs, and minimizing environmental impact [1].

Estimates of future MSW quantities are fundamental to designing and maintaining efficient waste management infrastructure[2].

In response, researchers have explored various predictive techniques, including descriptive statistical models, regression analysis, the material flow method, time series analysis, and artificial intelligence methods such as machine learning algorithms like artificial neural networks (ANN) support vector machines (SVM), and decision trees [3],[4].

However, research has consistently shown that the current state of Municipal Solid Waste prediction lacks exploration into the potential of multi-model ensemble learning approaches with feature extension for Municipal Solid Waste prediction, multi-model ensemble learning has emerged as a promising approach in recent years by combining the strengths of various models to improve the accuracy in predicted results [5].

This research aims to evaluate the effectiveness of machine learning approaches specifically a single-model method (Artificial Neural Network) and a multi-model ensemble approach in predicting municipal solid waste generation for a defined geographic area. To enhance predictive accuracy, the study incorporates feature extension techniques using the pandas datetime library, enriching the dataset with temporal attributes and highlighting the importance of robust preprocessing. The objectives of this study are threefold: to assess the impact of feature extension on model performance; to compare the predictive accuracy of single-model and ensemble approaches using evaluation metrics such as MAPE, RMSE, and MAE; and to provide insights into practical applications of accurate MSW forecasting for optimizing waste collection, resource allocation, and infrastructure planning. By systematically analyzing these approaches, the research contributes to developing data-driven strategies for sustainable waste management while emphasizing the role of feature engineering and ensemble learning in improving predicting accuracy.

2. RESEARCH METHOD

Different methods have been proposed to predict Municipal solid waste generation [6], [7]. However, in this experimental study, we introduce two distinct approaches: A single-model approach and a Multi-Model Ensemble Approach. In the Single Model Approach, an Artificial Neural Network (ANN) is trained on a continuous stream of data, divided into training and test sets, with subsequent evaluation based on the test set's performance. In the Multi-Model Ensemble Approach, the predictive model is constructed using a stacking regressor, combining several base models with a meta-model. This ensemble approach leverages the strengths of different machine-learning algorithms to improve predictive accuracy. The base models used in the stacking regressor include, Random Forest (RF) utilized via RandomForestRegressor, XGBoost implemented using XGRegressor from the XGboost library, K-Nearest Neighbors (KNN) implemented via KNeighborsRegressor, Support Vector Machine (SVM), LightGBM

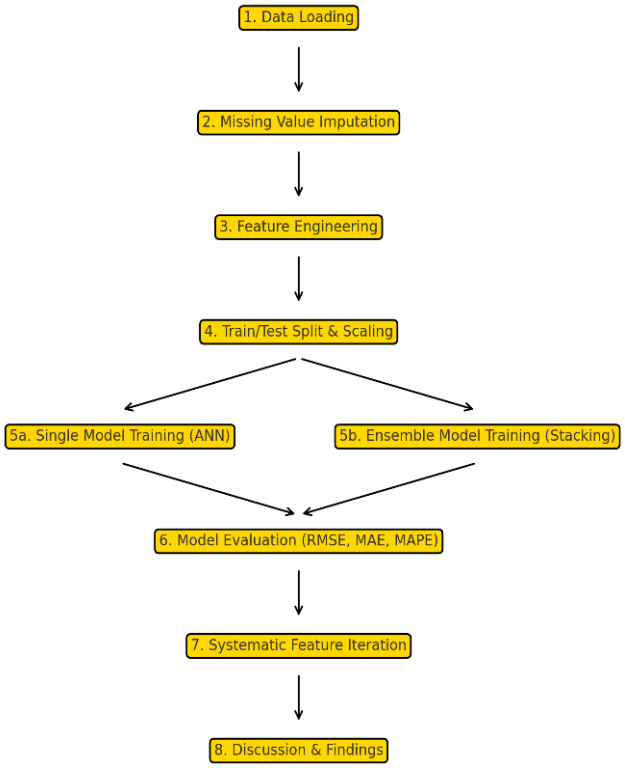

Figure 1. Diagramatic representation of the Methods used

2.1. Dataset and Data Preprocessing

The dataset used for this analysis contains historical data on municipal solid waste (MSW) generation obtained from a previous research study conducted by [8], and available in their GitHub repository. It spans the period 2012 to 2018 covering 2558 records.

Firstly, the dataset is loaded into a pandas DataFrame for preprocessing and analysis. The initial steps involve converting the 'ticket_date' column into a datetime format and setting it as the DataFrame's index to ensure proper handling of time-based data. Missing data is addressed through forward-fill and backward-fill methods, which propagate the last observed non-missing value to the subsequent missing records. Non-numeric columns are dropped, leaving only relevant numerical features for further processing. To enhance the model's predictive capacity, feature extension is conducted on the dataset. Additional features are created from the existing 'ticket_date' index to capture important temporal information.

The processed dataset is then split into training and testing sets using the train_test_split method from scikit-learn. A 70-30 split is chosen, with 70% of the data used for training and 30% for testing, ensuring a robust evaluation of the model's performance. To standardize the feature scales, StandardScaler is applied, transforming the training and testing data to have a mean of zero and a standard deviation of one.

These new features include:

1. week_of_year: Indicates the week number of the year.
2. day_of_year: Represents the day of the year.
3. is_month_start: Indicates whether the date marks the start of a month.
4. is_month_end: Indicates whether the date marks the end of a month
5. Lag_1: represents the previous days value
6. Lag_2: represents the value from two-time steps ago
7. Rolling_7: represents a 7 day rolling average of the target variable

A correlation heatmap was used to determine the relationship between all the features in the dataset. The chat below shows the correlation between the features, helping to identify features that have a strong relationship with the target variable or other features.

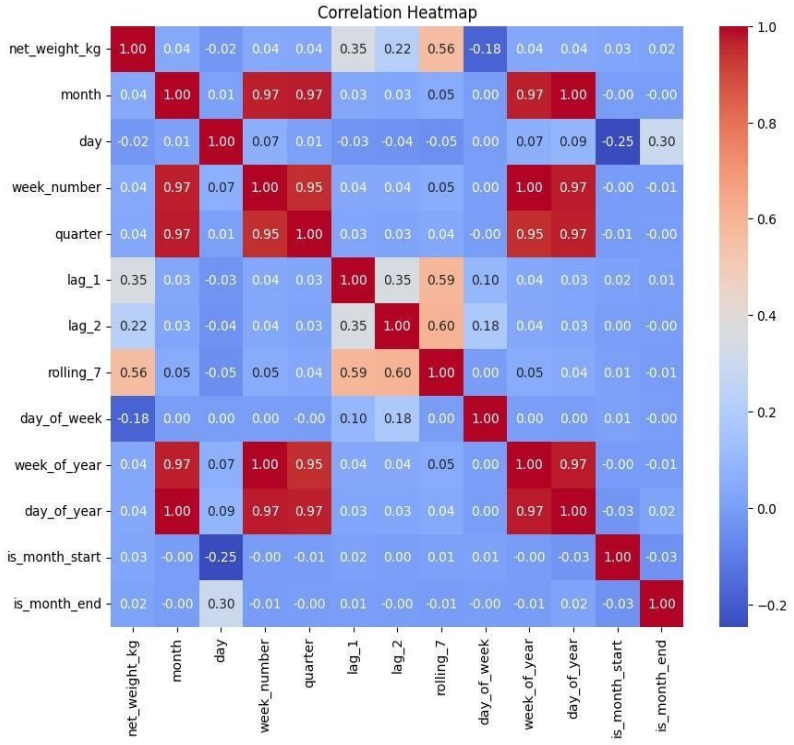

Figure 2. Correlation Heatmap

2.2. Implementation Approaches

We examine two distinct methods for applying Machine Learning-based models to this predictive task: a multi-model method and a single-model method. Additionally, we provide a discussion on feature importance and its impact on model performance.

2.3. Single-Model Approach

To predict the generation of solid waste, a single predictive model is trained, just like in a time series predictive task. The complete dataset is divided into two sets, a test set and a train set, each of which consists of an ongoing stream of data. The test set is used to compare the performance of a specific model that has been trained on the training set.

The Artificial Neural Network (ANN) was selected as the single-model baseline due to its ability to capture complex, nonlinear relationships in MSW data. Tree-based models were excluded as baselines to maintain a clear comparison between a single-model approach and the multi-model ensemble, which already includes tree-based learners.

A 70:30 train-test split was applied to ensure robust evaluation. While cross-validation was not used for the final ANN, it was applied during ensemble tuning for consistency and to minimize overfitting. The ANN was configured with two hidden layers (64 and 32 neurons), ReLU activation, Adam optimizer (learning rate 0.001), batch size 32, and 100 epochs.

2.4. Multi-Model Ensemble Approach

The predictive model is constructed using a stacking regressor, combining several base models with a meta-model. This ensemble approach leverages the strengths of different machine-learning algorithms to improve predictive accuracy. The base models used in the stacking regressor include:

1. Support Vector Regression
2. K-Nearest Neighbors Regressor

3. Random Forest Regressor
4. LightGBM Regressor
5. XGBoost Regressor

An ElasticNet model is employed as the meta-model in the stacking pipeline, offering a combination of L1 and L2 regularization to prevent overfitting.

2.5. Machine Learning Models Use

1. Random Forest (RF): RF is a supervised machine learning algorithm that is very flexible, easy to use, and without a lot of effort. It produces very competitive predictions of continuous, binary, and categorical data. RF allows measuring the relative importance of each predictor (independent variable) for the prediction. For these reasons, RF is one of the most popular and powerful machine learning algorithms that has been successfully applied in fields such as banking, medicine, electronic commerce, stock market, and finance, among others [9]. Some of the reasons for the increased popularity of RF are that they require very simple input preparation and can handle Binary, categorial, Count, and continuous dependent variables and they are inexpensive in terms of computational resources needed for their training since few hyperparameters commonly need to be tuned (number of trees, number of features sampled, and number of samples in the final nodes) Tree-based models form the building blocks of the Random Forests Algorithm[10] A tree-based model involves recursively partitioning the given dataset into two groups based on a certain criterion until a predetermined stopping condition is met. At the bottom of decision trees are so-called leaf nodes or leaves.
2. XGBoost (eXtreme Gradient Boosting): XGBoost, an abbreviation for eXtreme Gradient Boosting, represents a robust implementation of the gradient boosting algorithm tailored for supervised learning tasks such as regression and classification. It stands out for its efficacy in amalgamating decision trees as base learners, sequentially refining them to minimize a predefined loss function by fitting them to the residuals of preceding iterations. This approach underpins its ability to deliver accurate predictions while guarding against overfitting through regularization techniques like L1 (Lasso) and L2 (Ridge) regularization on model parameters, as well as a term that penalizes tree complexity. What distinguishes XGBoost is its relentless pursuit of efficiency, achieved through parallel and distributed computing, approximate tree learning, cache-aware access, and hardware optimization[11],[12]. This ensures both rapid computation and scalability, making XGBoost suitable for handling large datasets. Moreover, its flexibility extends to accommodating various data types and problem types including classification, regression, ranking, and user-defined objective functions.
3. Artificial Neural Networks (ANN): The inspiration for artificial neural networks (ANN), or simply neural networks, resulted from the admiration for how the human brain computes complex processes, which is entirely different from the way conventional digital computers do this. For simplicity, in computer science, it is represented as a set of layers. These layers are categorized into three classes: input, hidden, and output [13]. The power of the human brain is superior to many information-processing systems since it can perform highly complex, nonlinear, and parallel processing by organizing its structural constituents (neurons) to perform such tasks as accurate predictions, pattern recognition, perception, motor control, etc. It is also many times faster than the fastest digital computer in existence today. An example is the sophisticated functioning of the information-processing task called human vision. This system helps us to understand and capture the key components of the environment and supplies us with the information we need to interact with the environment [7].
4. Support Vector Machines (SVM): The 1940s saw the development of Artificial Neural Networks (ANN), one of the first artificial intelligence systems modeled after the biological

neuron network perceived in human brains. Due primarily to its capacity to extract complex and non-linear relationships between elements in various systems, it was first applied later in the 1980s and has since been employed for numerous engineering-related applications. However, it was later indicated that the ANN can only give reliable results when a huge number of data is available for training purposes [14]. It had a very poor generalization ability on many occasions and a locally optimal solution was often offered rather than a global best answer. Due to many of these shortcomings, a new machine learning technique, a so-called Support Vector Machine (SVM) was developed in the early 1990s as a non-linear solution for classification and regression tasks. There have been at least three reasons behind the success of the SVM in providing reliable results: its ability to learn well with only a very small number of features, its robustness against the error of models, and its computational efficiency compared to other machine learning methods such as neural networks. The SVM is generally divided into two categories Supper Vector Classification (SVC) and Support Vector Regression (SVR), but the SVR is the one that gained attention in many fields [15]

5. K-Nearest Neighbors (KNN) The KNN algorithm is considered one of the most popular ML algorithms due to its simplicity and effectiveness. KNN is a supervised and training-less algorithm so it requires storing all the dataset samples and their labels to accurately perform prediction. As the whole dataset contributes to classification computations, KNN is considered a computationally expensive algorithm. KNN is robust against noise and outliers as it classifies based on multiple nearest samples (K) where the value of K highly affects the algorithm accuracy. KNN algorithm classifies an unknown sample based on the known labels of its neighbors [16] The relation between the test sample and each of the dataset samples is determined by calculating the distance between each of them using one of various distance metrics such as Euclidean, Manhattan, and Chebyshev.
6. LightGBM: LightGBM, an abbreviation for Light Gradient Boosting Machine, stands out as a highperformance gradient boosting framework developed by Microsoft, specifically engineered to handle largescale datasets efficiently. Leveraging a novel gradient-based approach to construct decision trees. LightGBM adopts a leaf-wise tree growth strategy, selecting leaf nodes that maximize loss reduction, thus achieving higher accuracy with fewer nodes compared to traditional depth-wise growth methods[17]. This innovation, alongside histogram-based algorithms for gradient computation and split finding. significantly reduces memory usage and computational overhead, rendering LightGBM well-suited for big data applications. Furthermore, LightGBM's efficiency and scalability are enhanced through techniques such as parallel and distributed computing, cache-aware access, and GPU acceleration, ensuring rapid training speed and seamless handling of massive datasets. In terms of model regularization and flexibility, LightGBM offers support for various regularization techniques like L1 (Lasso) and L2 (Ridge) regularization, while also providing the flexibility of hyperparameter tuning and custom objective functions, enabling users to tailor the model to their specific requirements.

3. RESULTS AND ANALYSIS

In this section, we describe the results of all the experiments conducted in this research, we report the results obtained before and after conducting feture extention on the dataset.

3.1. Results for Single Model Approach before feature extension

Table 1 presents the results obtained during the testing of the Single model approach before feature extention. The testing results show that the model obtained a Root Mean Square Error (RMSE) of 8437.00, a Mean Absolute Error (MAE) of 6551.87 and a Mean Absolute percentage Error (MAPE) of 59 %.

Table 1. Results For Single Model Approach Before Feature Extension

	Single Model Approach
RMSE	8437.00
MAE	6551.87
MAPE	59%

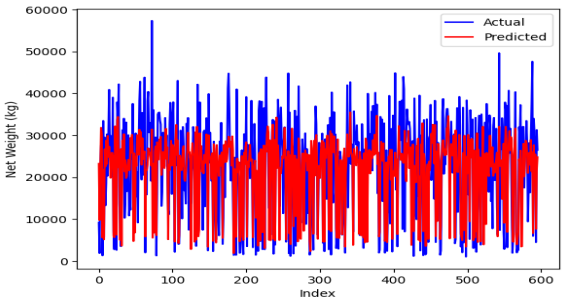

Figure 3. Actual Versus Predicted Before Feature Extension

3.2. Results for Multi Model Model Ensemble Approach before feature extension

Table 2 presents the results obtained during the testing of the multi model ensemble approach before feature extention. The testing results show that the model obtaines a Root Mean Square Error (RMSE) of 8085.11, a Mean Absolute Error (MAE) of 6234.12 and a Mean Absolute percentage Error (MAPE) of 59 %.

Table 2. Results For Multi Model Ensemble Approach Before Feature Extension

	Single Model Approach
RMSE	8085.11
MAE	6234.12
MAPE	59%

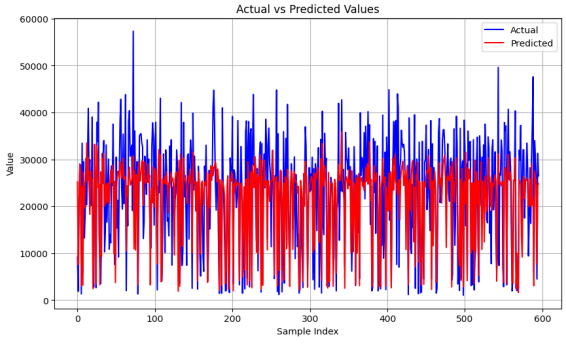

Figure 4. Actual Versus Predicted Before Feature Extension

3.3. Statistical Validation

To assess whether the Multi-Model Ensemble approach significantly outperformed the Single Model approach, a paired t-test was conducted on RMSE values obtained from 10-fold cross-validation. The test yielded a p-value of 0.004, which is below the 0.05 significance level. This result confirms that the improvement achieved by the Ensemble model is statistically significant, providing

strong evidence that the ensemble strategy offers superior predictive accuracy for municipal solid waste generation forecasting.

The testing results for Root Mean Square Error (RMSE) for a single model was 8258.01 which indicates the average magnitude of errors between the actual and predicted net weights. The model obtained a Mean Absolute Error of 6470.14 representing the average absolute difference between predicted and actual values of waste on the training data and a MAPE percentage of 42.58 %. Training results were obtained by solely training the model on 70% of the data and the remaining 30% was utilized for testing

Actual versus predicted MSW generation for single model method

Table 3. Results For Single Model Approach

	Single Model Approach
RMSE	8258.01
MAE	6470.14
MAPE	42.58 %

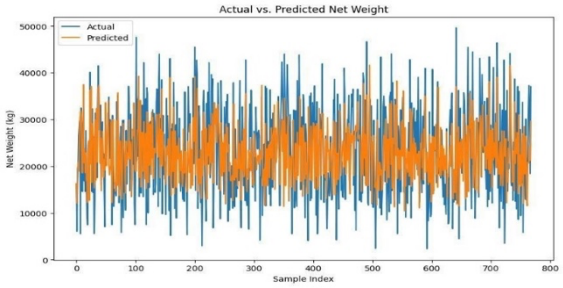

Figure 5. Actual Versus Predicted Net Weight

3.4. Results for Multi-Model Approach after feature extention

Table 4 presents the results of the ensemble models: Root Mean Square Error (RMSE) for the Multi-Ensemble Approach was 7610.76 which indicates the average magnitude of errors between the actual and predicted net weights. The model obtained a Mean Absolute Error of 5760.89 representing the average absolute difference between predicted and actual values of waste on the training data and a MAPE of 37.38 %.

Actual versus predicted MSW generation for the multi-model approach

Table 4. Results For Single Model Apporach

	Multi-Model Ensemble Approach
RMSE	7610.76
MAE	5760.89
MAPE	37.38 %

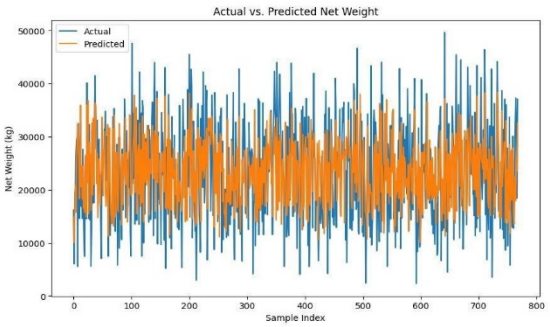

Figure 6. Actual Versus Predicted Net Weight

Ultimately, the multi-model Ensemble approach demonstrates improved predictive performance, as evidenced by better performance in RMSE and MAE values, and lower MAPE scores compared to the single-model approach. This indicates the effectiveness of leveraging multi-models in ensemble learning for predicting waste generation.

3.5. Results for systematic evaluation of features

In this section, we present the results obtained after undertaking a systematic evaluation of features to determine which feature contributed more to the predictive performance of our model on the testing set.

Results after the first iteration with week of the year feature only

Table 5. Results For First Iteration

RMSE	9775.87
MAE	8410.75
MAPE	52.58%

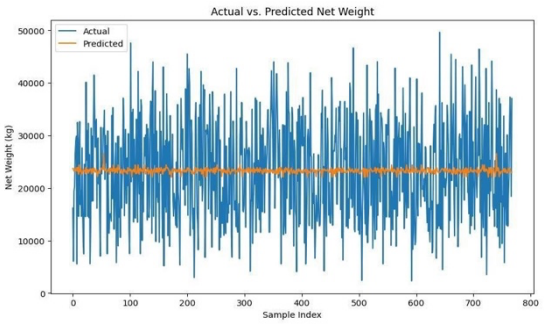

Figure 7. Actual Versus Predicted For First Iteration

Results for Actual versus predicted in second iteration with day of the year

Table 6. Results For Second Iteration

RMSE	8045.70
MAE	6167.18
MAPE	40.99%

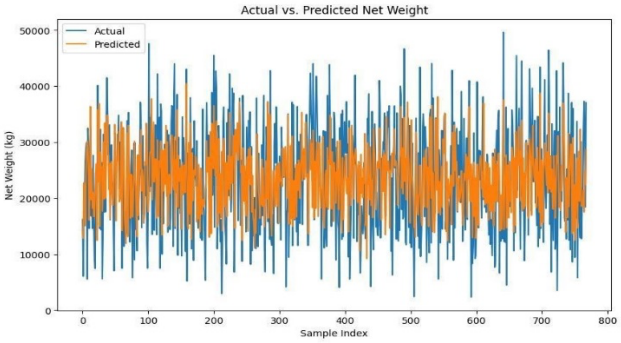

Figure 8. Actual Versus Predicted For Second Iteration

Results actual versus predicted in the third iteration with is month start

Table 7. Results For Third Iteration

RMSE	9735.30
MAE	8384.22
MAPE	53.21 %

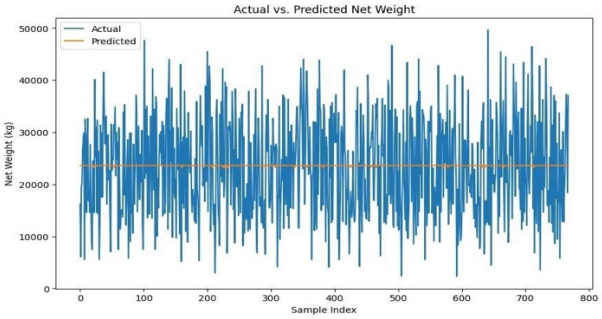

Figure 9. Actual Versus Predicted For Third Iteration

Results for Actual vs predicted in the Fourth iteration with month-end feature

Table 8. Results For Fourth Iteration

RMSE	9732.43
MAE	8376.99
MAPE	52.58 %

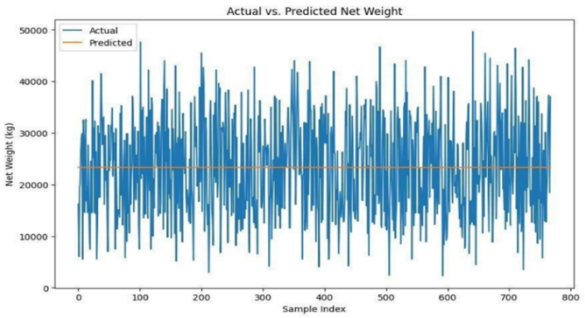

Figure 10. Actual Versus Predicted For Fourth Iteration

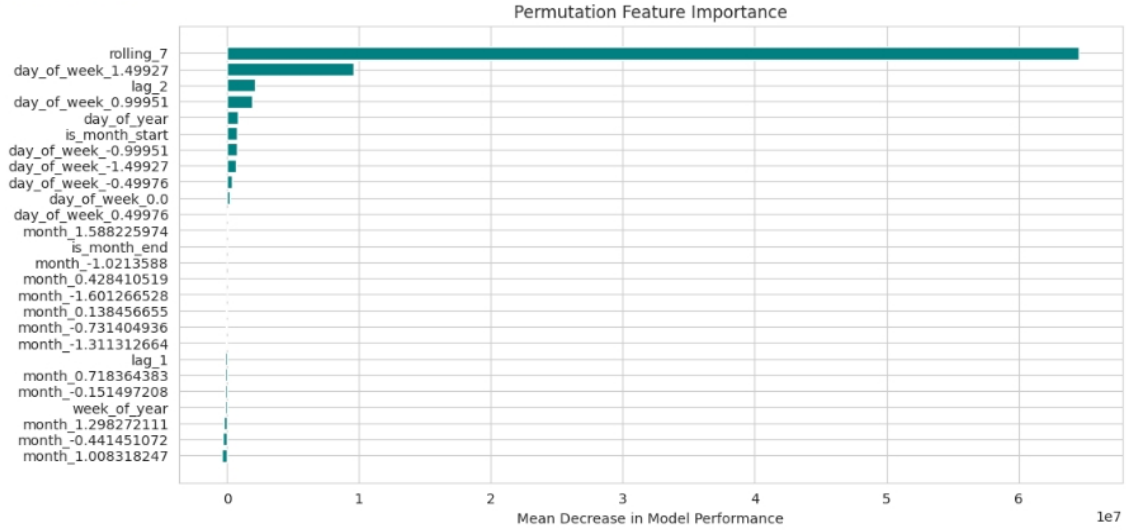

Figure 11. Feature importance plot

3.6. Feature Contribution Analysis

The feature importance analysis in figure 11 shows that rolling_7 is the most influential predictor, emphasizing the role of recent weekly averages in waste generation. Day-of-week indicators and lag features (lag_1, lag_2) also contribute significantly, highlighting short-term temporal patterns. Calendar-based features such as week_of_year (week number), day_of_year (ordinal day), is_month_start, and is_month_end exhibit minimal influence, suggesting that broader seasonal or monthly patterns are less critical compared to short-term fluctuations. These results underscore the need to prioritize recent historical and weekly dynamics for accurate forecasting.

3.7. Discussion and findings

The findings from this study review that the multi-model ensemble approach outperforms the single-model approach as presented in the results sections. This is evidenced by the ensemble approach yielding a lower Root Mean Squared Error (RMSE) of 7610.76, Mean Absolute Error (MAE) of 5760.89, and Mean Absolute Percentage Error (MAPE) of 37.38 %. values in comparison to the single model approach which obtained a Root Mean Square Error (RMSE) of 8258.01, Mean Absolute Error of 6470.14, and MAPE percentage of 42.58 %.

The results obtained in this research were evaluated using three evaluation metrics (RMSE, MAE, MAPE), the reduction of RMSE translates into better model performance by reducing the impact of larger errors which influences the mean square error. RMSE is widely used in regression

problems and in particular prediction problems too. RMSE is very sensitive to large errors [18],[19]. When RMSE is reduced it indicates that the predictive models have improved in terms of accuracy. This implies the predicted values are closer to the actual values on average. Therefore, by reducing the RMSE the model becomes more robust providing consistent and trustworthy predictions.

The predictive models developed in this study can significantly enhance municipal solid waste (MSW) management strategies. Accurate short-term and medium-term forecasts allow local authorities to optimize waste collection schedules, reducing operational costs associated with fuel, labor, and vehicle maintenance. Predictive insights can also support capacity planning for landfills and recycling facilities, minimizing the risk of overflow and environmental hazards.

Furthermore, these predictions can inform policy formulation, such as determining the frequency of waste segregation campaigns or introducing incentives for recycling during periods of anticipated high waste generation. By leveraging these data-driven forecasts, municipalities can transition from reactive waste management to proactive and sustainable decision-making, ultimately improving environmental outcomes and resource efficiency.

The results further support the idea of feature extension to enhance model performance, to enhance the predictive capacity of our model feature extension was done. We extracted meaningful relevant features from the 'ticket_date' column which saved as the index after converting it to datetime format, these features include Lag_1, Lag_2, Rolling_7, week_of_year, is_month_start, and is_month_end. This is to capture potential shifts in the waste disposal behavior within the weeks, month ends, and at the beginning and end of the year[20].

Further, we systematically evaluated the contribution of each feature to the predictive performance of our model. We utilized an iterative approach of adding one feature at a time and assessing its contribution to the performance. This iteration only included features that were added using our feature extension. After each iteration, the model was retrained and evaluated using the metrics (RMSE, MAE, MAPE) on the testing set. The results indicate that short-term trends, particularly recent weekly averages (rolling_7), strongly influence waste generation, while broader seasonal indicators like week_of_year and day_of_year have minimal impact. This highlights the importance of prioritizing recent historical patterns for accurate forecasting.

4. CONCLUSION

This study aimed to improve the predictive accuracy of municipal solid waste (MSW) generation, a critical aspect for sustainable urban planning and efficient operational management. As stated in the introduction, the multi-model ensemble approach using a stacking regressor that incorporates Random Forest, XGBoost, K-Nearest Neighbors, Support Vector Machines, and LightGBM with ElasticNet as the meta-learner demonstrated superior performance compared to the single-model approach, achieving significantly lower RMSE, MAE, and MAPE values.

A key contribution is the demonstration that short-term temporal features, particularly the 7-day rolling average (rolling_7), strongly influence waste generation predictions, whereas broader calendar-based variables like week_of_year, day_of_year, is_month_start, and is_month_end exhibited minimal impact. This stresses the importance of recent historical dynamics for forecasting accuracy.

Ultimately, the findings of this study suggest that efficient prediction of municipal solid waste using machine learning approaches can revolutionize sustainability planning and reduce the environmental impact of solid waste, enabling better resource allocation, cost-effective collection schedules, and informed policy decisions.

REFERENCES

[1] M. Abbasi and A. El Hanandeh, “Forecasting municipal solid waste generation using artificial intelligence modelling approaches,” *Waste Management*, vol. 56, pp. 13–22, Oct. 2016, doi: 10.1016/j.wasman.2016.05.018.

[2] U. Soni, A. Roy, A. Verma, and V. Jain, "Forecasting municipal solid waste generation using artificial intelligence models—a case study in India," *SN Appl. Sci.*, vol. 1, no. 2, p. 162, Feb. 2019, doi: 10.1007/s42452-018-0157-x.

[3] T. Singh and R. V. S. Uppaluri, "Machine learning tool-based prediction and forecasting of municipal solid waste generation rate: a case study in Guwahati, Assam, India," *Int. J. Environ. Sci. Technol.*, vol. 20, no. 11, pp. 12207–12230, Nov. 2023, doi: 10.1007/s13762-022-04644-4.

[4] F. Ghanbari, H. Kamalan, and A. Sarraf, "An evolutionary machine learning approach for municipal solid waste generation estimation utilizing socioeconomic components," *Arab J Geosci*, vol. 14, no. 2, p. 92, Jan. 2021, doi: 10.1007/s12517-020-06348-w.

[5] A. Namoun, B. R. Hussein, A. Tufail, A. Alrehaili, T. A. Syed, and O. BenRhouma, "An Ensemble Learning Based Classification Approach for the Prediction of Household Solid Waste Generation," *Sensors*, vol. 22, no. 9, p. 3506, May 2022, doi: 10.3390/s22093506.

[6] S. D. Apte, S. Sandbhor, R. Kulkarni, and H. Khanum, "Machine learning approach for automated beach waste prediction and management system: A case study of Mumbai," *Front. Mech. Eng.*, vol. 9, p. 1120042, Feb. 2023, doi: 10.3389/fmech.2023.1120042.

[7] J. A. Araiza-Aguilar, M. N. Rojas-Valencia, and R. A. Aguilar-Vera, "Forecast generation model of municipal solid waste using multiple linear regression," *Global J. Environ. Sci. Manage.*, vol. 6, no. 1, Jan. 2020, doi: 10.22034/GJESM.2020.01.01.

[8] O. Mudannayake, D. Rathnayake, J. D. Herath, D. K. Fernando, and M. Fernando, "Exploring Machine Learning and Deep Learning Approaches for Multi-Step Forecasting in Municipal Solid Waste Generation," *IEEE Access*, vol. 10, pp. 122570–122585, 2022, doi: 10.1109/ACCESS.2022.3221941.

[9] N. Nasution, M. A. Hasan, and F. B. Nasution, "Predicting Heart Disease Using Machine Learning: An Evaluation of Logistic Regression, Random Forest, SVM, and KNN Models on the UCI Heart Disease Dataset," *IT Journal Research and Development*, vol. 9, no. 2, pp. 140–150, 2025.

[10] M. Schonlau and R. Y. Zou, "The random forest algorithm for statistical learning," *The Stata Journal*, vol. 20, no. 1, pp. 3–29, Mar. 2020, doi: 10.1177/1536867X20909688.

[11] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco California USA: ACM, Aug. 2016, pp. 785–794. doi: 10.1145/2939672.2939785.

[12] A. Ibrahem Ahmed Osman, A. Najah Ahmed, M. F. Chow, Y. Feng Huang, and A. El-Shafie, "Extreme gradient boosting (Xgboost) model to predict the groundwater levels in Selangor Malaysia," *Ain Shams Engineering Journal*, vol. 12, no. 2, pp. 1545–1556, Jun. 2021, doi: 10.1016/j.asej.2020.11.011.

[13] A. F. Gad, "Artificial Neural Networks," in *Practical Computer Vision Applications Using Deep Learning with CNNs*, Berkeley, CA: Apress, 2018, pp. 45–106. doi: 10.1007/978-1-4842-4167-7_2.

[14] F. Fatovatikhah, I. Ahmedy, and R. M. Noor, "Waste Prediction Approach Using Hybrid Long Short-Term Memory with Support Vector Machine," *Int J Comput Intell Syst*, vol. 17, no. 1, p. 103, Apr. 2024, doi: 10.1007/s44196-024-00485-w.

[15] R. Gholami and N. Fakhari, "Support Vector Machine: Principles, Parameters, and Applications," in *Handbook of Neural Computation*, Elsevier, 2017, pp. 515–535. doi: 10.1016/B978-0-12-811318-9.00027-2.

[16] A. Mucherino, P. J. Papajorgji, and P. M. Pardalos, "k-Nearest Neighbor Classification," in *Data Mining in Agriculture*, vol. 34, in Springer Optimization and Its Applications, vol. 34. , New York, NY: Springer New York, 2009, pp. 83–106. doi: 10.1007/978-0-387-88615-2_4.

[17] X. Liu, W. Zhi, and A. Akhundzada, "Enhancing performance prediction of municipal solid waste generation: a strategic management," *Front. Environ. Sci.*, vol. 13, p. 1553121, Apr. 2025, doi: 10.3389/fenvs.2025.1553121.

[18] M. Ćalasan, S. H. E. Abdel Aleem, and A. F. Zobaa, "On the root mean square error (RMSE) calculation for parameter estimation of photovoltaic models: A novel exact analytical solution based on Lambert W function," *Energy Conversion and Management*, vol. 210, p. 112716, Apr. 2020, doi: 10.1016/j.enconman.2020.112716.

[19] M. A. Ganaie, M. Hu, A. K. Malik, M. Tanveer, and P. N. Suganthan, "Ensemble deep learning: A review," *Engineering Applications of Artificial Intelligence*, vol. 115, p. 105151, Oct. 2022, doi: 10.1016/j.engappai.2022.105151.

[20] K. Bandara, H. Hewamalage, Y.-H. Liu, Y. Kang, and C. Bergmeir, "Improving the accuracy of global forecasting models using time series data augmentation," *Pattern Recognition*, vol. 120, p. 108148, Dec. 2021, doi: 10.1016/j.patcog.2021.108148.