Engineering Applications

https://publish.mersin.edu.tr/index.php/enap

e-ISSN 2979-9201

Predicting Trip Purposes of Households in Makurdi Using Machine Learning: A Comparative Analysis of Decision Tree, CatBoost, and XGBoost Algorithms

Emmanuel Okechukwu Nwafor*[1], Folake Olubunmi Akintayo [2]

[1]*University of Ibadan, Department of Civil Engineering, Nigeria, engineerokeb@gmail.com*
[2] *University of Ibadan, Department of Civil Engineering, Nigeria,* fo.akintayo@ui.edu.ng

Cite this study: Nwafor, E.O., & Akintayo, F.O., (2024). Predicting Trip Purposes of Households in Makurdi Using Machine Learning: A Comparative Analysis of Decision Tree, CatBoost, and XGBoost Algorithms. Engineering Applications, 3 (3), 260-274.

Keywords
Trip Purpose
Machine Learning
Decision Tree
CatBoost
XGBoost

Research Article
Received:13.11.2024
Revised:14.11.2024
Accepted:18.12.2024
Published:30.12.2024

Abstract

This study explores the application of machine learning techniques for predicting trip purposes in Makurdi, Nigeria, utilizing three advanced algorithms: Decision Tree (DT), CatBoost, and XGBoost. The research aims to determine the most effective model for predicting household trip purposes based on demographic, socioeconomic, and travel data. Model performance was assessed using key metrics, including R-squared (R^2), Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE), revealing distinct strengths and weaknesses among the models. CatBoost demonstrated the highest R^2 score of 73%, indicating its efficacy in capturing variance in trip purposes, despite a higher MAE (0.353) and RMSE (0.850), which suggest potential for larger prediction errors. XGBoost, with an R^2 score of 72% and the lowest RMSE of 0.545, exhibited a balanced performance, providing accurate predictions with minimal error. The Decision Tree model, while acceptable with an R^2 of 68%, MAE of 0.314, and RMSE of 0.615, ranked lower in predictive accuracy. The findings advocate for the use of XGBoost as the most reliable model for this task. Future research directions include hyperparameter optimization and the investigation of ensemble methods to enhance predictive accuracy.

1. Introduction

The rapid urbanization of cities worldwide has necessitated the development of innovative tools for analysing and predicting travel behaviour to support efficient, sustainable transportation systems. Cities like Makurdi, Nigeria, are grappling with the challenges posed by increasing population density, limited public transit infrastructure, and the need for comprehensive urban planning [1-4]. Understanding how households make travel decisions such as choosing destinations, modes of transportation, and trip purposes is crucial for developing policies that enhance accessibility, reduce congestion, and promote sustainability [5-7]. To this end, predictive models are becoming essential for urban planners, providing actionable insights from large-scale, complex data on urban mobility [8-10].

In urban transportation studies, trip purpose prediction is a vital aspect of travel demand modelling, providing insight into daily travel patterns within a metropolitan region [11-13]. Traditionally, the four-step transportation model has been used to understand travel demand and behaviour. However, its reliance on aggregated data often limits its ability to capture complex, individual travel behaviours and respond flexibly to rapid changes in urban environments [14-20]. This shortcoming has led researchers to explore data-intensive,

machine learning-based models that offer fine-grained predictions based on demographic and socioeconomic characteristics, facilitating more accurate analyses of individual travel choices [21]. By focusing on specific trip purposes (e.g., work, education, shopping, leisure, business), machine learning models offer planners precise and adaptable forecasting tools that help guide infrastructure investments, policy development, and transit design [22-26].

In recent years, machine learning techniques have increasingly been adopted in transportation research, with algorithms like Decision Trees, CatBoost, and XGBoost showing significant promise [27-29]. Decision Tree algorithms are widely valued for their interpretability and ability to capture hierarchical patterns in travel behaviour, making them particularly useful for predicting trip purposes based on demographic and household characteristics [30-32]. Unlike traditional regression models, Decision Trees handle non-linear relationships and high-dimensional datasets, providing clear, rule-based classifications that urban planners can easily interpret [33-36]. CatBoost and XGBoost, on the other hand, are part of a class of gradient boosting algorithms that iteratively improve prediction accuracy by combining the predictions of multiple weak learners [37-40]. CatBoost, specifically designed to handle categorical data effectively, minimizes prediction bias, while XGBoost has been celebrated for its computational efficiency, especially in processing large datasets [41-44].

Integrating these algorithms in urban transportation modelling offers several advantages. Firstly, they support dynamic and responsive modelling, where changes in travel behaviour patterns can be identified and incorporated in real time [45-48]. For instance, CatBoost's unique treatment of categorical variables, common in transportation data (e.g., mode choice, income level), makes it suitable for analysing the impacts of socio-economic factors on trip purposes, thereby capturing nuanced travel behaviours [49-51]. Furthermore, XGBoost's computational efficiency and accuracy have been successfully applied in high-dimensional transportation studies, including origin-destination modelling and mode choice predictions [52-55]. These attributes make both algorithms suitable for deployment in rapidly growing cities like Makurdi, where timely, data-driven decisions are essential for addressing mobility needs.

Additionally, as urban centres adopt the concept of "smart cities," integrating machine learning into transportation planning aligns with broader efforts to digitize urban infrastructure [56-58]. Smart city initiatives, which emphasize real-time data analysis and predictive modelling, benefit from machine learning's ability to process diverse data streams, such as household surveys, transit system data, and social media feeds, to generate insights for transportation planning [59-61]. In this regard, Makurdi's adoption of a strategic digital city framework, where AI-driven insights can guide transportation infrastructure development and policy, illustrates how machine learning is reshaping urban mobility solutions in emerging economies [62-64].

Moreover, with advancements in machine learning, the accuracy and robustness of trip purpose prediction models have improved, making them highly suitable for diverse urban settings [65-66]. For example, while Decision Trees are straightforward and interpretable, their predictive power is bolstered when combined with gradient boosting techniques like those in CatBoost and XGBoost, which iteratively refine predictions [67-68]. By implementing these models, this research aims to develop and compare three machine learning techniques—Decision Tree, CatBoost, and XGBoost—for their effectiveness in predicting trip purposes in Makurdi. The outcome will provide urban planners and policymakers with a validated toolset for designing responsive, sustainable transportation systems tailored to the specific needs of Makurdi's population.

This study addresses the critical gap in trip purpose prediction for emerging cities, contributing to the body of research that leverages machine learning for urban transportation modeling. By focusing on Makurdi, a city representative of the broader challenges faced by rapidly urbanizing regions in Africa, this research will highlight how machine learning techniques can be adapted for localized, scalable solutions in transportation planning. Through this, Makurdi could serve as a case study for applying advanced machine learning models in urban mobility, providing insights that may benefit similar cities worldwide.

2. Material and Method

2.1. Description of Study Area

Makurdi, Nigeria, serves as the capital of Benue State and is geographically positioned between latitudes 7°37'60"N to 7°50'20"N and longitudes 80°19'30"E to 80°40'20"E, at an elevation of 93 meters above sea level. The town is predominantly drained by the Benue River, which bifurcates it into the northern and southern sections, interconnected by two bridges. The economic activities of residents in the Makurdi metropolis primarily encompass civil service, commerce, and agrarian peasantry. The population of Makurdi metropolis is estimated at

500,797 individuals [69], with the highest density located in the High Level, Wadata, and Wurukum districts [70]. The geographical location of the study area is illustrated in Fig. 1.

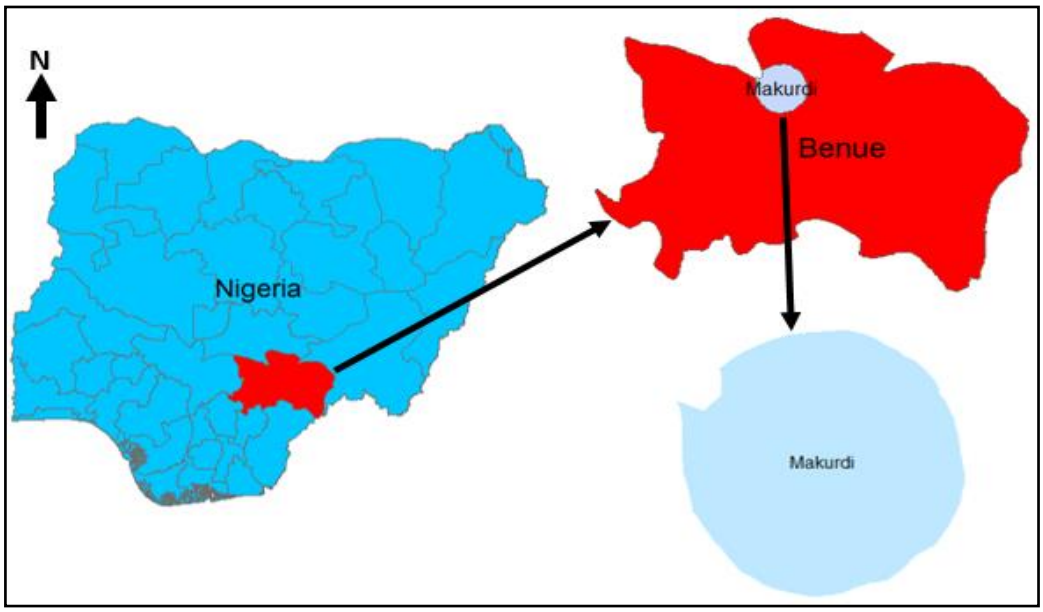

Figure 1. Location of Study Area

2.2 Source of Data

Data for this study were collected through a household questionnaire interview survey conducted in the study area from January 2021 to December 2022. The study area was delineated into nine Traffic Analysis Zones (TAZ), corresponding to the geopolitical council wards of the city, which include Bar, Walumayo, Fiide, Modern Market, Wadata/Ankpa, Central South, Clerk/Market, North Bank, and North Bank 2. Methodologically, this research adopts a case study approach employing both qualitative and quantitative techniques [71-73]. Revealed preference questionnaires were distributed to households within the Makurdi metropolis to gather data on travel demand in relation to the demographic characteristics of the households. Various data collection methods were utilized, including online platforms (Google Forms, Survey Monkey, and WhatsApp), email, and in-person interviews at residences. The systematic random sampling technique was implemented for the travel survey, whereby every third household along designated streets within the study locations was selected for participation. The questionnaire comprised items specifically designed to elicit socioeconomic data and current travel information from respondents. Key attributes and data types essential for the study included gender, age, economic status, the number of household members, the number of vehicles available for use by household members, the number and types of driving licenses held by household members, along with other relevant household characteristics, which served as dependent variables.

2.3 Sample Size

The sample size for this study was determined using the formula proposed by [74], which represents a modification of [75] formula. This approach was employed to derive an optimal sample size representative of the study area.

$$n = \frac{\mathrm{N}}{1 + \mathrm{N}\mathcal{E}^2} \tag{1}$$

Where:

n = Minimum returned sample size

N = Population size

e = The degree of accuracy express as proportion

ρ = The number of standard deviations that would include all possible values in the range

t = t-value for the selected alpha level or confidence level at 95%

2.4 Machine Learning (ML) Models for Trip Purpose Predictive Modelling

2.4.1 Modelling using Python Programming

This study utilized three advanced machine learning models: Decision Tree, CatBoost, and XGBoost to develop predictive algorithms based on comprehensive household demographic and trip information data collected through an extensive survey. Each model was implemented in Python within the Google Colaboratory environment, leveraging Python's robust data science libraries.

The dataset underwent thorough pre-processing steps, including exploratory data analysis, correlation analysis, and checks for missing values and outliers. Descriptive statistical summaries were generated to assess data distribution and prepare for optimized model training and validation while the target variables are the trip purposes such as home-based work (HBW), home-based education (HBE), home-based shopping (HBS), home-based leisure (HBL), non home-based (NHB) and home-based other trip (HBO).

2.4.2 Model Training and Validation

The dataset employed for model development was structured to include all relevant input parameters for efficient modeling and predictive analysis. For optimal training and evaluation, the data was partitioned into training and testing subsets, allocating 80% to model training and 20% to testing, executed using the train_test_split function from Python's Scikit-Learn (Sklearn) library. The modeling process utilized a suite of Python libraries, including Pandas for data manipulation, Seaborn and Matplotlib for data visualization, Numpy for numerical operations, Joblib for model serialization, and Google Colab for cloud-based computation. Fig. 2 presents the code snippet demonstrating the importation of essential libraries, alongside key functions such as mean_squared_error, train_test_split, and StandardScaler applied in model construction and validation.

```
# Step 1: Import necessary libraries
import pandas as pd
import numpy as np
import seaborn as sns
from google.colab import files
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeRegressor, export_text, plot_tree
from sklearn.metrics import r2_score, mean_squared_error, mean_absolute_error
import matplotlib.pyplot as plt
```

Figure 2. Importing Important Python Libraries, Modules and Functions

2.4.3 Model Evaluation and Visualisations

The models' performance was rigorously assessed using key evaluation metrics, including Mean Squared Error (MSE), Mean Absolute Error (MAE), and R-squared (R^2) values, providing quantitative insights into model accuracy and error distribution. Additionally, visualization techniques were employed to illustrate comparative model performance, enhancing interpretability and allowing for a detailed analysis of prediction accuracy across models.

3. Results and Discussion

3.1. Summary of Dataset for Modelling

The dataset utilized for modeling comprises household and trip-related information from a total of 1,802 households, where each row represents a unique household sample, and each column corresponds to one of the 25 collected characteristics. In this dataset structure, 19 variables were defined as independent (or feature) variables, with the remaining 6 classified as target (or dependent) variables. Features encompass household demographics and travel characteristics such as household size, age, occupation, gender, income level, vehicle ownership, and mode of transportation used. The target variables specifically represent trip purposes, including categories like home-based work (HBW), home-based education (HBE), and non-home-based (NHB) travel.

Key insights from the data revealed that the average household size is 3.65, with a mean employment rate of 2.01 individuals per household, an average of 1.43 students per household, and gender distribution averaging 2.03 males and 1.62 females per household. For the purpose of modeling, 7 of these features were selected as the input variables to predict the 6 distinct trip purposes, as illustrated in the summary statistics in Table 1. This setup enabled the predictive model to efficiently utilize the household and demographic data to classify trip purposes accurately

Table 1. Descriptive Statistics of Dataset Obtained from Questionnaire Survey

Variables	mean	std	min	25%	50%	75%	max
Household Size	3.65	1.28	1	3	4	4	12
Employed	2.01	0.79	0	2	2	2	6
Student	1.43	1.1	0	1	1	2	6
Unemployed	0.23	0.46	0	0	0	0	2
Income	72961	35971	0	50000	65000	86000	250000
Car Ownership	0.42	0.51	0	0	0	1	3
Motorcycle Ownership	0.23	0.42	0	0	0	0	1
HBW	4.2	1.74	0	4	4	4	12
HBE	2.78	2.17	0	2	2	4	10
HBS	3.11	1.68	0	2	4	4	8
HBL	0.99	1.3	0	0	0	2	8
HBO	1.04	1.12	0	0	0	2	4
NHB	0.82	1.08	0	0	0	2	4

Source: Survey Data

3.2 Pearson's Correlation Analysis

A correlation analysis was conducted for each pair of variables within the dataset using Python's built-in correlation functions to quantify relationships. Given the high dimensionality of the correlation matrix, a heatmap (Fig. 3) was generated to visually represent the results, facilitating clearer interpretation. The analysis revealed correlation coefficients across the dataset ranging from -0.3 to 1.0, with a color-coded bar on the right side of the heatmap indicating the intensity of correlation values for each variable pair. This visualization effectively highlights patterns of linear dependency within the dataset.

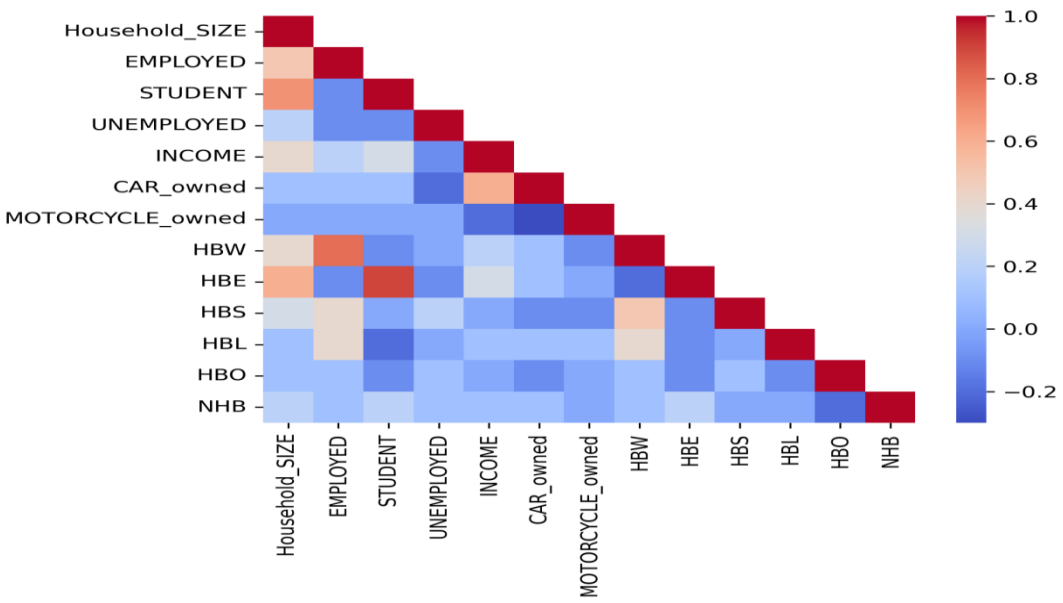

Figure 3. Heat Map of Correlation Analysis

3.3 Modelling

Following pre-processing, the dataset was split into independent (feature) and dependent (target) variables, then partitioned into 80% for training and 20% for testing. Subsequently, each of the three models was instantiated and trained on the designated training subset to build predictive capabilities.

3.3.1 Decision Tree Model

The optimal architecture for the Decision Tree model was determined by systematically evaluating multiple tree structures, each parameterized by varying Max_Depth values. Model accuracy and error were quantified using R-squared (R^2) and Mean Absolute Error (MAE) as performance metrics. A Python code snippet detailing this selection process is illustrated in Fig. 4, with corresponding results summarized in Table 2.

```
for i in range (1,26):
    dt_model = DecisionTreeRegressor(max_depth=i, min_samples_split=
                                     3, random_state=42)
    dt_model.fit(X_train, y_train)

    # Step 3: Model Evaluation
    y_pred = dt_model.predict(X_test)
    r2 = r2_score(y_test, y_pred)
    mae = mean_absolute_error(y_test, y_pred)
    mse = mean_squared_error(y_test, y_pred)
    print('max_depth = ', i)
    print(f'R-squared value: {r2}')
    print(f'Mean Absolute Error: {mae}')
    #print(f'Mean Squared Error: {mse}')
    print()
```

Figure 4. Python Code Snippet

Table 2. Searching for the Optimal Value of 'Max_Depth' Parameter

Decision Tree Max_Depth	Model Testing Accuracy	Mean Absolute Error
1	0.1105	1.131
2	0.2012	1.075
3	0.2665	0.958
4	0.3382	0.876
5	0.3994	0.772
6	0.4631	0.688
7	0.5267	0.6
8	0.5585	0.536
9	0.5866	0.484
10	0.617	0.424
11	0.64192	0.391
12	0.6681	0.344
13	0.6736	0.327
14	0.672	0.323
15	0.6737	0.321
16	0.6815	0.313
17	0.6815	0.313
18	0.6815	0.313

Due to the specific data characteristics in this study, the prediction accuracy of the Decision Tree model increased, while error decreased, as the Max_Depth parameter was incremented—up to a depth of 16, where accuracy and error metrics stabilized, indicating convergence. Thus, a Max_Depth of 16 was selected as optimal for this model. Fig. 5 illustrate the structure of the Decision Tree model at depths 2, for interpretative clarity. The full tree structure, extending to 16 levels, captures all variables and decision paths, ensuring comprehensive predictions at each terminal node, though it is too detailed to display fully in this document.

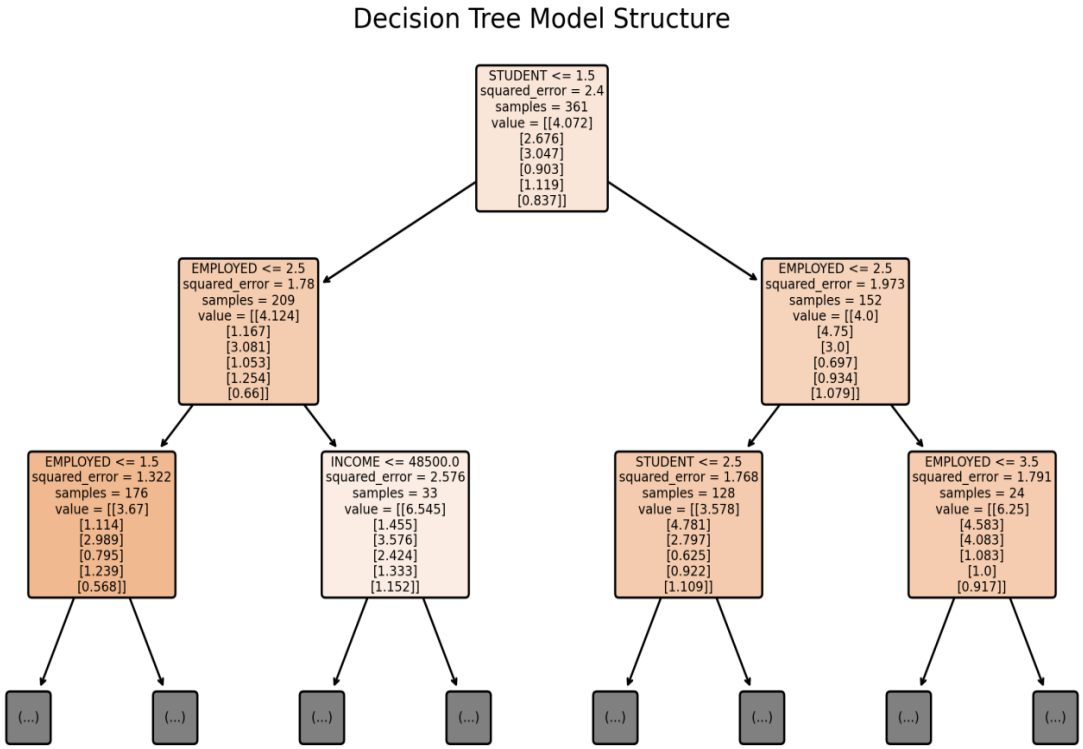

Figure 5. Decision Tree Model Structure (From the Root Mode to the Second Level)

3.3.2 CatBoost and XGBoost Models

This study applied multiple advanced techniques to enhance the accuracy of the CatBoost and XGBoost models, including feature engineering (logarithmic transformations and data scaling), cross-validation, hyperparameter tuning, model regularization, and increasing the number of trees within the models. Given that optimal solutions are data-dependent, these approaches were systematically tested, and those yielding the best performance were selected for the final models. Hyperparameter tuning, in particular, proved most effective in refining model performance. Tables 3 and 4 list the key hyperparameters adjusted for CatBoost and XGBoost to achieve optimal accuracy, while Figs. 6 and 7 display corresponding Python code snippets for model building and training.

Table 3. CatBoost and XGBoost Hyper-Parameters Tuned in this Study

CatBoost	Tuning Range	XGBoost	Tuning Range
Iterations	Positive integers	Number of trees (N_estimators)	Positive integers
Learning rate	0.01 to 0.4	Learning rate	0.01 to 0.4
Depth	4 to 10	Max_depth	4 to 10
L2_leaf_reg	1 to 10	Reg_alpha	1 to 10
		Reg_lambda	1 to 10

The GridSearchCV method, implemented in Python, was employed to systematically explore various combinations of hyperparameter values. This technique facilitates the identification of parameter configurations that optimize model performance while mitigating the risk of overfitting. By tuning the hyperparameters, the method aims to achieve the highest possible accuracy. The optimal parameter settings identified through this process are presented in Table 4.

Table 4. Best Hyper-Parameters by GridSearchCV for the Models

CatBoost	Values	**XGBoost**	Values
Iterations	1000	N_estimators	1000
Learning rate	0.1	Learning rate	0.2
Depth	6	Max_depth	6
L2_leaf_reg	1	Reg_alpha	1
		Reg_lambda	2

```
from catboost import CatBoostRegressor
from sklearn.metrics import r2_score, mean_squared_error, mean_absolute_error

# Initialize list to store CatBoost models
catboost_models = []
# Initialize list to store predictions
train_predictions = []
test_predictions = []

# Loop through each target variable column
for i in range(6):  # Assuming 6 target columns
    # Initialize CatBoost model
    model = CatBoostRegressor(iterations=1000, depth=6, learning_rate=0.1,
                              l2_leaf_reg=1, random_state=42)

    # Fit the model on the training data
    model.fit(X_train, y_train.iloc[:, i], verbose=0)
```

Figure 6. Building the CatBoost Model With Optimal Hyperparameters

```
from xgboost import XGBRegressor
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

# XGBoost Model Building
# Initialize XGBoost model
xgb_model = XGBRegressor(n_estimators=100, learning_rate=0.2,max_depth=6,
                         reg_alpha=1, reg_lambda=2,verbosity=0)

# Fit the model to the training data
xgb_model.fit(X_train, y_train)
```

Figure 7. Building the XGBoost Model With Optimal Hyperparameters

3.4 Model Performance Evaluation and Visualisations

The performance of the trained models was subsequently assessed using R-squared, Mean Absolute Error (MAE), and Mean Squared Error (MSE), with corresponding Python code snippets illustrated in Figs. 8 to 10 Additionally, the models underwent validation by applying them to predict outcomes on previously unseen data (the 20% test dataset). The performance metrics were computed using the same evaluation criteria, and the results are detailed in Tables 5 to 7 for each model.

```
# Evaluate the DT Model Performance in Training and Testing

# Predict the target values on the training dataset
y_pred = dt_model.predict(X_train).round(0)
# R-squared, Mean Absolute Error and Mean Squared Error
r2 = r2_score(y_train, y_pred)
mae = mean_absolute_error(y_train, y_pred)
rmse = mean_squared_error(y_train, y_pred,squared=False)
print(f'R-squared value: {r2}')
print(f'Mean Absolute Error: {mae}')
print(f'Root Mean Squared Error: {rmse}')

# Predict the target variables on the test set
y_pred_dt = dt_model.predict(X_test)
# Calculate testing the R2 score, MAE, and MSE
dt_r2_score = r2_score(y_test, y_pred_dt)
dt_mae = mean_absolute_error(y_test, y_pred_dt)
dt_rmse= mean_squared_error(y_test, y_pred_dt, squared=False)
# Print the results
print('Decision tree prediction:')
print('R2 score:', dt_r2_score)
print('MAE:', dt_mae)
print('RMSE', dt_rmse)
```

Figure 8. Code to Evaluate the Performance of the DT Model

```
# Evaluate performance metrics
train_r2 = r2_score(y_train, train_predictions_array)
train_mae = mean_absolute_error(y_train, train_predictions_array)
train_rmse = mean_squared_error(y_train, train_predictions_array,
                                squared = False)
test_r2 = r2_score(y_test, test_predictions_array)
test_mae = mean_absolute_error(y_test, test_predictions_array)
test_rmse = np.sqrt(mean_squared_error(y_test, test_predictions_array,
                                       squared=False))
## Print performance metrics
print('Training Performance:')
print(f'R-squared value: {train_r2}')
print(f'Mean Absolute Error: {train_mae}')
print(f'Root Mean Squared Error: {train_rmse}')

print('\nTesting Performance:')
print(f'R-squared value: {test_r2}')
print(f'Mean Absolute Error: {test_mae}')
print(f'Root Mean Squared Error: {test_rmse}')
```

Figure 9. Code to Evaluate the Performance of the CatBoost Model

```
# Model Evaluation
# Predict target variables for training and testing sets
xgb_train_pred = xgb_model.predict(X_train)
xgb_test_pred = xgb_model.predict(X_test)

# Calculate evaluation metrics
xgb_train_mae = mean_absolute_error(y_train, xgb_train_pred)
xgb_test_mae = mean_absolute_error(y_test, xgb_test_pred)
xgb_train_rmse = mean_squared_error(y_train, xgb_train_pred)
xgb_test_rmse = mean_squared_error(y_test, xgb_test_pred)
xgb_train_r2 = r2_score(y_train, xgb_train_pred)
xgb_test_r2 = r2_score(y_test, xgb_test_pred)

# Print evaluation metrics and hyperparameters
print("XGBoost Model:")
print("Train MAE:", xgb_train_mae)
print("Test MAE:", xgb_test_mae)
print("Train RMSE:", xgb_train_rmse)
print("Test RMSE:", xgb_test_rmse)
print("Train R2 Score:", xgb_train_r2)
print("Test R2 Score:", xgb_test_r2)
print("XGBoost Hyperparameters:", xgb_model.get_params())
```

Figure 10. Code to Evaluate the Performance of the XGBoost Model

Table 5. Performance of the Decision Tree model for Training and Validation

Performance Metrics	Training	Validation
R-squared value	0.841978	0.681508
Mean Absolute Error (MAE)	0.187717	0.313950
Root Mean Squared Error (RMSE)	0.301527	0.615197

Table 6. Performance of the CatBoost model for Training and Validation

Performance Metrics	Training	Validation
R-squared value	0.8439267	0.7313342
Mean Absolute Error (MAE)	0.2579396	0.3528039
Root Mean Squared Error (RMSE)	0.5395861	0.8503091

Table 7. Performance of the XGBoost model for Training and Validation

Performance Metrics	Training	Validation
R-squared value	0.851961	0.722995
Mean Absolute Error (MAE)	0.224566	0.331785
Root Mean Squared Error (RMSE)	0.283789	0.545492

The accuracy results indicate that all three models: Decision Tree, CatBoost, and XGBoost—demonstrated satisfactory performance in predicting trip purposes, achieving training accuracies of 84.19%, 84.39%, and 85.2%, respectively. Following the testing and validation phase, the models exhibited accuracies of 68.1%, 73.1%, and 72.3%, respectively. All models displayed acceptably low prediction error values for both training and testing scenarios. The high accuracy rates observed in both training and validation phases suggest that the models did not overfit the data, instead effectively learning the underlying patterns, thereby ensuring their utility for future predictions. Consequently, these models can be considered reliable for estimating household trip purpose decisions in future applications. To further analyze and compare the models' effectiveness for prospective applications, their R-squared values and error metrics on the validation dataset were visualized, as shown in Fig. 11.

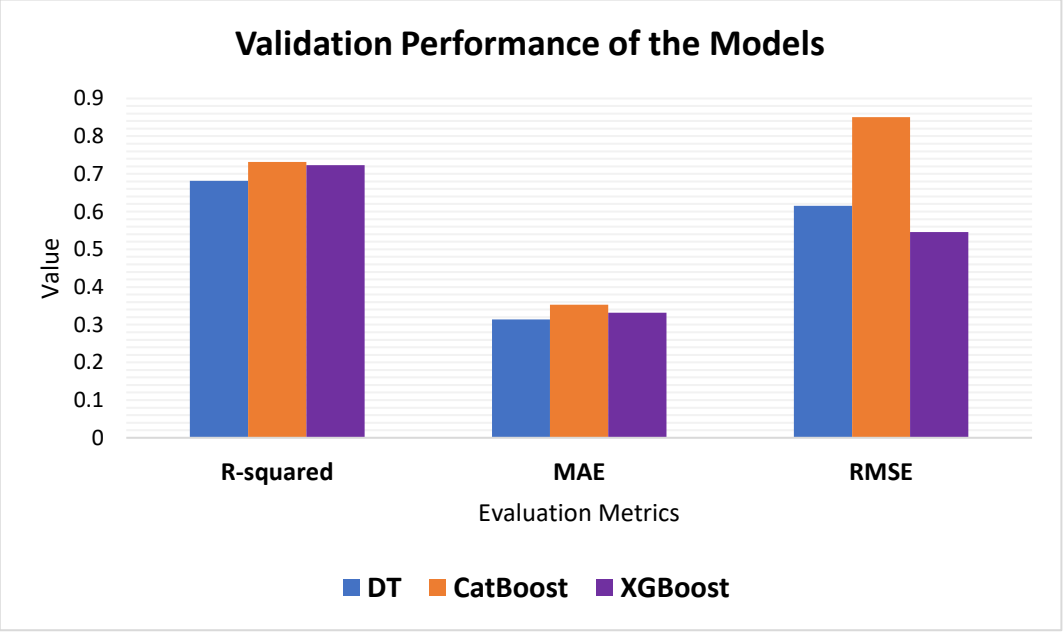

Figure 11. Validation Performance of DT, CatBoost, XGBoost

Based on the results and visualizations: The Decision Tree model attained an R^2 score of 0.681508, signifying that approximately 68.15% of the variance in trip purposes can be accounted for by this model. The Mean Absolute Error (MAE) was recorded at 0.31395, while the Root Mean Squared Error (RMSE) was 0.615197. These error metrics indicate that, although the model demonstrates reasonable performance, there remains potential for enhancement in minimizing prediction errors. CatBoost exhibited the highest R^2 score of 0.7313342 among the three models, indicating its superior capability in capturing the variance in trip purposes. Nevertheless, its MAE of

0.3528039 and RMSE of 0.8503091 reveal greater error values compared to the Decision Tree model. The relatively elevated RMSE suggests that, while CatBoost generally performs accurately, it may yield larger errors in specific predictions. XGBoost achieved a competitive R^2 score of 0.722995, closely trailing CatBoost. Its MAE was measured at 0.331785, and it recorded the lowest RMSE of 0.545492 among the models. This performance indicates that XGBoost not only accounts for a substantial portion of the variance in trip purposes but also delivers the most precise predictions with minimal error.

4. Discussion

This study investigated the utilization of advanced machine learning models—namely, Decision Tree (DT), CatBoost, and XGBoost—for the prediction of trip purposes based on household and travel data. The models were rigorously evaluated using key performance metrics: R-squared (R^2), Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE). The results delineate the distinct strengths and weaknesses inherent to each model. The analysis indicates that CatBoost, with the highest validation R^2 score of 0.7313342, is the most effective model for capturing the variance in trip purposes. However, its relatively elevated MAE of 0.3528039 and RMSE of 0.8503091 imply that it may generate larger prediction errors in certain instances when compared to the other models. XGBoost, achieving an R^2 score of 0.722995 alongside the lowest RMSE of 0.545492, emerges as the most balanced model, providing accurate predictions with minimal error. In contrast, the Decision Tree model, while performing satisfactorily with an R^2 of 0.681508, MAE of 0.31395, and RMSE of 0.615197, lags behind both CatBoost and XGBoost in terms of predictive accuracy and error reduction. In conclusion, although all three models exhibit potential for predicting trip purposes from household data, XGBoost is recommended due to its superior balance of high explanatory power and low prediction errors, rendering it the most reliable option for this task. Future research should concentrate on further hyperparameter optimization for these models and the exploration of ensemble techniques to enhance predictive performance.

Acknowledgement

The authors would like to thank the editorial team and prospective reviewers for considering our manuscript for review. We appreciate the opportunity to share our research with the *Journal of Engineering Application* and value any feedback that will help refine and improve the study.

Funding

This research received no external funding.

Author contributions:

Conceptualization: [Emmanuel Okechukwu Nwafor, Folake Olubunmi Akintayo]; Methodology: [Emmanuel Okechukwu Nwafor, Folake Olubunmi Akintayo]; Formal analysis and investigation: [Emmanuel Okechukwu Nwafor, Emmanuel Okechukwu Nwafor]; Writing - original draft preparation: [Emmanuel Okechukwu Nwafor]; Writing - review and editing: [Emmanuel Okechukwu Nwafor, Folake Olubunmi]; Supervision: [Folake Olubunmi Akintayo, Emmanuel Okechukwu Nwafor]

Conflicts of interest

The authors declare no conflicts of interest.

References

1. Tao, X., Cheng, L., Zhang, R., Chan, W. K., Chao, H., & Qin, J. (2023). Towards Green Innovation in Smart Cities: Leveraging Traffic Flow Prediction with Machine Learning Algorithms for Sustainable Transportation Systems. Sustainability, 16(1), 251.
2. Ang, K. L. M., Seng, J. K. P., Ngharamike, E., & Ijemaru, G. K. (2022). Emerging technologies for smart cities' transportation: geo-information, data analytics and machine learning approaches. ISPRS International Journal of Geo-Information, 11(2), 85.
3. Anagnostopoulos, T. (2021). A predictive vehicle ride sharing recommendation system for smart cities commuting. Smart Cities, 4(1), 177-191.
4. Musa, A. A., Malami, S. I., Alanazi, F., Ounaies, W., Alshammari, M., & Haruna, S. I. (2023). Sustainable Traffic Management for Smart Cities Using Internet-of-Things-Oriented Intelligent Transportation Systems (ITS): Challenges and Recommendations. Sustainability, 15(13), 9859.

5. Gallo, M., & Marinelli, M. (2020). Sustainable mobility: A review of possible actions and policies. Sustainability, 12(18), 7499.
6. Dingil, A. E., Rupi, F., & Esztergár-Kiss, D. (2021). An integrative review of socio-technical factors influencing travel decision-making and urban transport performance. Sustainability, 13(18), 10158.
7. Guo, Y., & Peeta, S. (2020). Impacts of personalized accessibility information on residential location choice and travel behavior. Travel Behaviour and Society, 19, 99-111.
8. Wang, Y., Currim, F., & Ram, S. (2022). Deep learning of spatiotemporal patterns for urban mobility prediction using big data. Information Systems Research, 33(2), 579-598.
9. He, W., & Chen, M. (2024). Advancing Urban Life: A Systematic Review of Emerging Technologies and Artificial Intelligence in Urban Design and Planning. Buildings, 14(3), 835.
10. Khan, A. F., & Ivan, P. (2023). Integrating Machine Learning and Deep Learning in Smart Cities for Enhanced Traffic Congestion Management: An Empirical Review. J. Urban Dev. Manag, 2(4), 211-221.
11. Karami, Z., & Kashef, R. (2020). Smart transportation planning: Data, models, and algorithms. Transportation Engineering, 2, 100013.
12. Alsaleh, N., & Farooq, B. (2021). Interpretable data-driven demand modelling for on-demand transit services. Transportation Research Part A: Policy and Practice, 154, 1-22.
13. Yang, B., Tian, Y., Wang, J., Hu, X., & An, S. (2022). How to improve urban transportation planning in big data era? A practice in the study of traffic analysis zone delineation. Transport policy, 127, 1-14.
14. Park, K., Sabouri, S., Lyons, T., Tian, G., & Ewing, R. (2020). Intrazonal or interzonal? Improving intrazonal travel forecast in a four-step travel demand model. Transportation, 47, 2087-2108.
15. Waghmare, A., Yadav, G., & Tiwari, K. (2022). Four Step Travel Demand Modeling for Urban Transportation Planning. Sci. Eng. Technol., 5, 1254.
16. Lwin, W. Y., Yoon, B. J., & Lee, S. M. (2024). Exercising The Traditional Four-Step Transportation Model Using Simplified Transport Network of Mandalay City in Myanmar. Journal of the Society of Disaster Information, 20(2), 257-269.
17. Miller, E. J. (2020). Travel demand models, the next generation: Boldly going where no-one has gone before. In Mapping the Travel Behavior Genome (pp. 29-46). Elsevier.
18. Mukherjee, J., & Kadali, B. R. (2022). A comprehensive review of trip generation models based on land use characteristics. Transportation Research Part D: Transport and Environment, 109, 103340.
19. Hasnine, M. S., & Nurul Habib, K. (2021). Tour-based mode choice modelling as the core of an activity-based travel demand modelling framework: A review of state-of-the-art. Transport Reviews, 41(1), 5-26.
20. Huang, Y., Gao, L., Ni, A., & Liu, X. (2021). Analysis of travel mode choice and trip chain pattern relationships based on multi-day GPS data: A case study in Shanghai, China. Journal of transport geography, 93, 103070.
21. Heidari, A., Navimipour, N. J., & Unal, M. (2022). Applications of ML/DL in the management of smart cities and societies based on new trends in information technologies: A systematic literature review. Sustainable Cities and Society, 85, 104089.
22. Abid, M. T., Aljarrah, N., Shraa, T., & Alghananim, H. M. (2024). Forecasting and managing urban futures: Machine learning models and optimization of urban expansion. Asian Journal of Civil Engineering, 25(6), 4673-4682.
23. AlKhereibi, A. H., Wakjira, T. G., Kucukvar, M., & Onat, N. C. (2023). Predictive machine learning algorithms for metro ridership based on urban land use policies in support of transit-oriented development. Sustainability, 15(2), 1718.
24. Kayıran, H. F., & Şahmeran, U. (2022). Development of individualized education system with artificial intelligence Fuzzy logic method. Engineering Applications, 1 (2), 137-144
25. Zela, K., & Saliaj, L. (2023). Forecasting through neural networks: Bitcoin price prediction. Engineering Applications, 2 (3), 218-224
26. Kayıran, H. F. (2022). The function of artificial intelligence and its sub-branches in the field of health. Engineering Applications, 1 (2), 99-107
27. Li, J., Wang, X., Yang, X., Zhang, Q., & Pan, H. (2024). Analyzing freeway safety influencing factors using the CatBoost model and interpretable machine-learning framework, SHAP. Transportation research record, 2678(7), 563-574.
28. RK, P., M. AboRas, K., & Youssef, A. (2024). Application of an ensemble CatBoost model over complex dataset for vehicle classification. Plos one, 19(6), e0304619.
29. Behboudi, N., Moosavi, S., & Ramnath, R. (2024). Recent Advances in Traffic Accident Analysis and Prediction: A Comprehensive Review of Machine Learning Techniques. arXiv preprint arXiv:2406.13968.
30. Koushik, A. N., Manoj, M., & Nezamuddin, N. (2020). Machine learning applications in activity-travel behaviour research: a review. Transport reviews, 40(3), 288-311.
31. Wang, S., Mo, B., Hess, S., & Zhao, J. (2021). Comparing hundreds of machine learning classifiers and discrete choice models in predicting travel behavior: an empirical benchmark. arXiv preprint arXiv:2102.01130.

32. Wu, W., Xia, Y., & Jin, W. (2020). Predicting bus passenger flow and prioritizing influential factors using multi-source data: Scaled stacking gradient boosting decision trees. IEEE Transactions on Intelligent Transportation Systems, 22(4), 2510-2523.
33. Tekouabou, S. C. K., Diop, E. B., Azmi, R., Jaligot, R., & Chenal, J. (2022). Reviewing the application of machine learning methods to model urban form indicators in planning decision support systems: Potential, issues and challenges. Journal of King Saud University-Computer and Information Sciences, 34(8), 5943-5967.
34. Mienye, I. D., & Jere, N. (2024). A Survey of Decision Trees: Concepts, Algorithms, and Applications. IEEE Access.
35. Sebt, M. V., Sadati-Keneti, Y., Rahbari, M., Gholipour, Z., & Mehri, H. (2024). Regression Method in Data Mining: A Systematic Literature Review. Archives of Computational Methods in Engineering, 1-20.
36. Vincent, R. R., Sakthivel, E., Kumari, M., Nisha, F., & Rohini, A. (2024). Machine Learning for Geospatial Analysis: Enhancing Spatial Understanding and Decision-Making. In Ethics, Machine Learning, and Python in Geospatial Analysis (pp. 168-195). IGI Global.
37. Zhang, L., & Jánošík, D. (2024). Enhanced short-term load forecasting with hybrid machine learning models: CatBoost and XGBoost approaches. Expert Systems with Applications, 241, 122686.
38. Hancock, J. T., & Khoshgoftaar, T. M. (2020). CatBoost for big data: an interdisciplinary review. Journal of big data, 7(1), 94.
39. Bentéjac, C., Csörgő, A., & Martínez-Muñoz, G. (2021). A comparative analysis of gradient boosting algorithms. Artificial Intelligence Review, 54, 1937-1967.
40. Demir, S., & Sahin, E. K. (2023). An investigation of feature selection methods for soil liquefaction prediction based on tree-based ensemble algorithms using AdaBoost, gradient boosting, and XGBoost. Neural Computing and Applications, 35(4), 3173-3190.
41. Joshi, A., Saggar, P., Jain, R., Sharma, M., Gupta, D., & Khanna, A. (2021). CatBoost—An ensemble machine learning model for prediction and classification of student academic performance. Advances in Data Science and Adaptive Analysis, 13(03n04), 2141002.
42. Kinnander, M. (2020). Predicting profitability of new customers using gradient boosting tree models: Evaluating the predictive capabilities of the XGBoost, LightGBM and CatBoost algorithms.
43. Zheng, Q., Yu, C., Cao, J., Xu, Y., Xing, Q., & Jin, Y. (2024). Advanced Payment Security System: XGBoost, CatBoost and SMOTE Integrated. arXiv preprint arXiv:2406.04658.
44. Hasan, B., Shaikh, S. A., Khaliq, A., & Nadeem, G. (2024). Data-Driven Decision-Making: Accurate Customer Churn Prediction with Cat-Boost. The Asian Bulletin of Big Data Management, 4(02), Science-4.
45. Almahdi, A., Al Mamlook, R. E., Bandara, N., Almuflih, A. S., Nasayreh, A., Gharaibeh, H., ... & Jamal, A. (2023). Boosting Ensemble Learning for Freeway Crash Classification under Varying Traffic Conditions: A Hyperparameter Optimization Approach. Sustainability, 15(22), 15896.
46. Almeida, M. R. M. R. (2024). Hybrid Failure Prognosis Approach combining Data-Driven and Knowledge-Based Methods.
47. Zhen, H., & Yang, J. J. (2024). Analyzing the importance of network topology in AADT estimation: insights from travel demand models using graph neural networks. Transportation, 1-38.
48. Abouelela, M., Lyu, C., & Antoniou, C. (2023). Exploring the Potentials of Open-Source Big Data and Machine Learning in Shared Mobility Fleet Utilization Prediction. Data Science for Transportation, 5(2), 5.
49. Senthilkumar, V. (2023). Enhancing House Rental Price Prediction Models for the Swedish Market: Exploring External features, Prediction intervals and Uncertainty Management in Predicting House Rental Prices.
50. Santamato, V., Tricase, C., Faccilongo, N., Iacoviello, M., Pange, J., & Marengo, A. (2024). Machine Learning for Evaluating Hospital Mobility: An Italian Case Study. Applied Sciences, 14(14), 6016.
51. Olaleye, O. (2024). Machine Learning and Stochastic Simulation for Inventory Management (Doctoral dissertation, Massachusetts Institute of Technology).
52. Martín-Baos, J. Á., López-Gómez, J. A., Rodriguez-Benitez, L., Hillel, T., & García-Ródenas, R. (2023). A prediction and behavioural analysis of machine learning methods for modelling travel mode choice. Transportation research part C: emerging technologies, 156, 104318.
53. Hu, S. (2023). A Big-Data-Driven Framework for Spatiotemporal Travel Demand Estimation and Prediction (Doctoral dissertation, University of Maryland, College Park).
54. Liu, Y., Miller, E., & Habib, K. N. (2022). Detecting transportation modes using smartphone data and GIS information: evaluating alternative algorithms for an integrated smartphone-based travel diary imputation. Transportation Letters, 14(9), 933-943.
55. Chen, Y., Geng, M., Zeng, J., Yang, D., Zhang, L., & Chen, X. M. (2023). A novel ensemble model with conditional intervening opportunities for ride-hailing travel mobility estimation. Physica A: Statistical Mechanics and its Applications, 628, 129167.
56. Wu, P., Zhang, Z., Peng, X., & Wang, R. (2024). Deep learning solutions for smart city challenges in urban development. Scientific Reports, 14(1), 5176.
57. Wolniak, R., & Stecuła, K. (2024). Artificial Intelligence in Smart Cities—Applications, Barriers, and Future Directions: A Review. Smart Cities, 7(3), 1346-1389.

58. Nikitas, A., Michalakopoulou, K., Njoya, E. T., & Karampatzakis, D. (2020). Artificial intelligence, transport and the smart city: Definitions and dimensions of a new mobility era. Sustainability, 12(7), 2789.
59. Sarker, I. H. (2022). Smart City Data Science: Towards data-driven smart cities with open research issues. Internet of Things, 19, 100528.
60. Soomro, K., Bhutta, M. N. M., Khan, Z., & Tahir, M. A. (2019). Smart city big data analytics: An advanced review. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 9(5), e1319.
61. França, R. P., Monteiro, A. C. B., Arthur, R., & Iano, Y. (2021). An overview of the machine learning applied in smart cities. Smart cities: A data analytics perspective, 91-111.
62. Lukic Vujadinovic, V., Damnjanovic, A., Cakic, A., Petkovic, D. R., Prelevic, M., Pantovic, V., ... & Bodolo, I. (2024). AI-Driven Approach for Enhancing Sustainability in Urban Public Transportation. Sustainability, 16(17), 7763.
63. Mayuranathan, M., Nahar, G., Vijayakumar, A., Mamodiya, U., & Babu, D. M. (2024). Sustainable Business Models for Smart City Using Artificial Intelligence Techniques. In Navigating the Circular Age of a Sustainable Digital Revolution (pp. 263-294). IGI Global
64. Van Hoang, T. (2024). Impact of integrated artificial intelligence and internet of things technologies on smart city transformation. Journal of Technical Education Science, 19(Special Issue 01), 64-73.
65. Boukerche, A., & Wang, J. (2020). Machine learning-based traffic prediction models for intelligent transportation systems. Computer Networks, 181, 107530.
66. Yuan, T., da Rocha Neto, W., Rothenberg, C. E., Obraczka, K., Barakat, C., & Turletti, T. (2022). Machine learning for next-generation intelligent transportation systems: A survey. Transactions on emerging telecommunications technologies, 33(4), e4427.
67. Tavakoli, F. (2023). Dataset Creation and Imbalance Mitigation in Big Data: Enhancing Machine Learning Models for Forest Fire Prediction (Master's thesis, University of Waterloo).
68. Yeung, C., Bunker, R., Umemoto, R., & Fujii, K. (2024). Evaluating soccer match prediction models: a deep learning approach and feature optimization for gradient-boosted trees. Machine Learning, 1-24.
69. National Population Commission, (2006). National Population Census of Federal Republic of Nigeria Official Gazette, 96 (2).
70. Abah, R.C., (2012). Causes of Seasonal Flooding in Flood Plains: A Case of Makurdi, Northern Nigeria. Intl. J. Envtal Studies, Vol. 69, No. 6, Pp. 904-912.
71. Morin, J. F., Olsson, C., & Atikcan, E. O. (2021). Research methods in the social sciences: an A-Z of key concepts. Oxford University Press.
72. Nachmias, D. & Nachmias, C. F. (2014). Research Methods in the Social Sciences, 8th ed. Worth Publishers.
73. Yin, R. K. (2017). Case study research and applications: design and methods, 6th ed. Sage Publications.
74. Adam, A. M. (2020). Sample Size Determination in Survey Research. *Journal of Scientific Research and Reports*, 26(5), 90–97.
75. Yamane, Y. (1967). Mathematical Formulae for Sample Size Determination.