ISSN 2776-8007

Classifying Vehicle Categories Based on Technical Specifications Using Random Forest and SMOTE for Data Augmentation

Dwi Sugianto[1,*], Tri Wahyuningsih[2]

[1,2]*Doctorate Program of Computer Science, Universitas Kristen Satya Wacana, Jawa Tengah, Indonesia*

(Received: June 5, 2025; Revised: July 15, 2025; Accepted: October 25, 2025; Available online: December 7, 2025)

Abstract

This study investigates the application of machine learning for classifying vehicles based on their technical specifications using the Random Forest algorithm. The objective was to create a robust classification model capable of categorizing vehicles into six distinct classes: Hybrid, SUV, Sedan, Sports, Truck, and Wagon. The analysis was conducted using a comprehensive dataset that included features such as engine size, horsepower, weight, and fuel efficiency, along with the target variable, vehicle class. To address the issue of class imbalance, the Synthetic Minority Over-sampling Technique (SMOTE) was applied to balance the training data. The results showed that the model performed particularly well in classifying Sedans, achieving a perfect recall and high F1-score, while struggling with underrepresented classes like Hybrid and Wagon. Despite applying SMOTE, the model's performance for minority classes remained suboptimal, highlighting the challenges associated with highly imbalanced datasets. The study contributes to the field of vehicle classification by demonstrating the use of Random Forest for such tasks and providing insights into the challenges posed by imbalanced class distributions. The findings underscore the importance of feature selection, especially regarding numerical attributes such as horsepower and engine size, in improving classification accuracy. However, the study also identified limitations, including potential biases in the dataset and the difficulty in improving performance for minority vehicle classes. Future research should explore alternative algorithms like XGBoost or deep learning models, and consider expanding the dataset to include more diverse vehicle types. The practical implications of this work are significant for vehicle market segmentation, offering valuable insights for manufacturers, dealerships, and analysts seeking to optimize vehicle classification and improve market targeting strategies.

Keywords: Vehicle Classification, Random Forest, SMOTE, Machine Learning, Market Segmentation

1. Introduction

The classification of vehicles based on technical specifications is increasingly critical in a variety of fields, including transportation management, infrastructure planning, and safety enforcement. Vehicle classification methods enable the efficient collection and processing of traffic data, which can subsequently enhance roadway design and management. As urban areas grow and vehicle traffic intensifies, the demand for effective vehicle classification systems that utilize advanced technologies has surged. Traditional methods of vehicle classification, which primarily depended on human observation and basic sensory input, have now evolved into complex systems combining machine learning, computer vision, and multi-sensor integration, thereby enhancing accuracy and efficiency in understanding traffic dynamics and behaviors. In recent years, the integration of smart technologies has revolutionized vehicle classification, allowing for the categorization of vehicles based on a multitude of features, including size, weight, and shape [1]. For instance, Velisavljević et al. [1] demonstrated a wireless magnetic sensor network capable of achieving an 88.37% accuracy rate in classifying five vehicle classes from extensive experimental data, indicating the effectiveness of modern methodologies over traditional approaches. The study highlights that the enhanced cost-effectiveness and energy efficiency of these systems make them viable for widespread deployment in vehicle classification tasks. Similarly, Atouf et al. [2] developed a real-time vehicle detection system that incorporates shadow removal and classifies vehicles based on their distinctive dimensions and features utilizing specified machine learning techniques. This dual approach

[*] **Corresponding author: Dwi Sugianto (dwisugianto@outlook.com)**

DOI: https://doi.org/10.47738/ijaim.v5i4.113

not only improves classification outcomes but also provides a scalable solution that can adapt to different urban environments.

Intelligent Transportation Systems (ITS) further exemplify the significance of vehicle classification technologies as they aggregate multiple data sources ranging from video footage captured by visible light and thermal cameras to acoustic sensors to improve traffic management and surveillance capabilities [3]. These systems are essential for categorizing vehicles during traffic accidents, facilitating timely responses and enhancing public safety. Moreover, Rajab et al. [4] emphasized the importance of accurately classifying vehicle types, including motorcycles, which present unique challenges for classification systems today [5]. Such granularity in classification is crucial for developing safety measures and optimizing roadway designs. The flow of traffic management is also contingent upon the capabilities of vehicle classification systems to accommodate the complexities of modern urban transport networks. Wei et al. [6] noted that the increasing volume of vehicles on the road necessitates sophisticated vehicle type classification not only for traffic management but also for electronic toll collections and autonomous driving scenarios. In this context, identifying emergency vehicles such as ambulances and fire engines can significantly enhance emergency response times and efficient traffic flow.

Accurate classification of vehicles remains a pressing challenge in the rapidly evolving landscape of ITS and urban traffic management. Various approaches exist to tackle this issue, each leveraging different sources of data and machine learning techniques to enhance classification accuracy amidst the complex dynamics of vehicular movement. The significance of classifying vehicles correctly cannot be overstated, as it influences traffic flow regulation, enhances safety measures, informs infrastructure planning, and optimizes urban mobility systems. Traditional image classification methods often face limitations due to environmental factors and variability in vehicle appearances. Noh and Jeon present an innovative solution by introducing Local Size-Specific Classifiers (LSCs), which dynamically adjust to normalized sizes according to contextual scene information [7]. This approach contrasts with conventional methods that rely on preset dimensions and offers enhanced reliability and interpretability in diverse traffic scenes. By incorporating both appearance cues and spatial location data for vehicle detection, LSCs effectively address several challenges associated with vehicle classification under varied environmental conditions.

Moreover, the advent of low-cost mm-Wave radar technology enables sophisticated monitoring capabilities. Abedi et al. [8] demonstrate the ability of radar systems to capture vital spatial information regarding vehicle occupancy, thereby augmenting traditional classification techniques with new dimensions of data. By utilizing reflectance signals processed through machine learning classifiers like Support Vector Machines (SVM), the accuracies achieved in classification tasks can significantly improve, addressing issues previously associated with reliance solely on optical systems. A more comprehensive approach to vehicle classification involves integrating multi-sensor data fusion strategies. Li et al. [9] illustrate the effectiveness of deep learning architectures in conjunction with multiple sensory inputs such as LiDAR and optical imagery [10]. This method not only enhances robustness against variable environmental conditions but also contributes to the overall accuracy of vehicle categorization, crucial for automated toll collection systems and traffic management applications. Through the deployment of Convolutional Neural Networks (CNNs), the authors provide evidence supporting the potential for increased classification efficacy, particularly as it pertains to the challenges presented by simple optical recognition in diverse conditions.

The goal of utilizing Random Forest and the Synthetic Minority Over-sampling Technique (SMOTE) for classifying vehicle categories is rooted in addressing the common challenges posed by class imbalance and improving the accuracy of predictive models in the realm of vehicular classification. As the demand for precise traffic monitoring and control escalates with the growth of smart city initiatives and automated systems, ensuring high classification performance becomes indispensable for various applications, including traffic safety, urban planning, and autonomous driving systems. Random Forest is a highly favored machine learning algorithm, particularly in classification tasks due to its robustness, capability to handle large datasets, and inherent feature selection mechanisms [11]. Its ensemble nature offers enhanced accuracy by combining the predictions of multiple decision trees, thus mitigating the likelihood of overfitting and improving generalization on unseen data. According to Kaya et al. [11], the performance of the Random Forest algorithm is significantly beneficial when leveraging diverse feature sets, enabling effective discrimination among different vehicle categories based on various attributes such as size, type, and operational status. In scenarios where training datasets are imbalanced where certain vehicle types are underrepresented compared to others the effectiveness of classification algorithms, including Random Forest, may be compromised. This challenge leads to a propensity for classification bias, where predictions are skewed towards the majority class. To counter this limitation, the application of SMOTE emerges as a strategic intervention. SMOTE works by synthetically generating additional instances of the minority class, thereby balancing the dataset and facilitating a more equitable learning process for classifiers. By augmenting the dataset with synthesized examples that maintain the characteristics of underrepresented classes, SMOTE enhances the model's ability to recognize and accurately classify minority vehicle categories [11].

This study is highly relevant to the vehicle market analysis and classification, as it provides a novel approach to accurately categorizing vehicles based on their technical specifications, such as engine size, horsepower, and drivetrain type. By utilizing machine learning techniques like Random Forest and applying SMOTE for data augmentation, the research enhances classification accuracy, particularly in handling imbalanced datasets. The findings can help industry stakeholders, such as car manufacturers, dealerships, and market analysts, better understand vehicle segmentation, improve inventory management, and make informed decisions based on vehicle attributes, thereby offering valuable insights for strategic planning and marketing within the automotive sector.

2. Literature Review

2.1. Vehicle Classification Studies

The utilization of machine learning in vehicle classification has proliferated in recent years, resulting in a diverse array of methodologies, algorithms, and technologies across various studies. This literature review presents significant works that explore these developments, particularly emphasizing the implementation of different machine learning techniques and their effectiveness in addressing the challenges of vehicle classification. Zhou et al. [12] discuss the incorporation of machine learning in automotive intelligence, including intelligent vehicle classification systems applied to autonomous vehicles. Their research highlights the significance of artificial intelligence in enhancing operational efficiency and real-time monitoring of vehicle states, thereby showcasing the growing intersection between IoT technology and vehicular systems [12], [13]. This synergy underscores the trend toward creating increasingly automated and intelligent transportation networks.

Abedi et al. [8] emphasize the use of low-cost mm-Wave radar technology in vehicle monitoring systems. Their study evaluates how features derived from radar signals specifically, spatial information related to occupancy within vehicles can be effectively harnessed through machine learning classifications including Random Forests, K-Nearest Neighbors (KNN), and SVM. The application of these algorithms demonstrates the adaptive capabilities of machine learning in dynamic environments, thus contributing meaningful insights into the development of in-vehicle automated systems [8]. Ahmad et al. [14] offer a distinct perspective by applying seismic fingerprinting as a means of vehicle auto-classification. The study highlights how modern vehicle detection systems leverage machine learning techniques alongside up-to-date sensing technologies. The findings suggest that the effectiveness of classification heavily relies not only on the algorithms employed but also on the specificity of the sensor technologies and the contextual parameters of the operational environment. Their work articulates the importance of sensor diversity as a proxy for operational efficiency and classification accuracy [14].

Ramasamy et al. [15] introduce a hybrid Deep Boltzmann Functional Link Network aimed at enhancing classification performance. By comparing their method against traditional algorithms like SVMs and neural networks, they provide empirical evidence supporting the superiority of their approach across several classification problems, including vehicle classification. Such investigations into hybrid models emphasize an ongoing quest to optimize algorithm performance beyond conventional techniquesc [15]. In a complementary study, Kleyko et al. [16] focus on a feature-free data approach for vehicle classification using roadside sensors. Their methodology eschews traditional feature extraction, employing a data mining technique known as "data smashing." This novel approach leverages raw signals, purportedly streamlining the classification process and reducing the need for extensive preprocessing a significant departure from conventional methodologies and indicative of evolving perspectives on sensor data and machine learning integration [16].

2.2. SMOTE in Classification

The SMOTE has emerged as a pivotal approach for addressing the challenges associated with class imbalance in various classification tasks across multiple domains. Class imbalance often leads to degradation in the performance of machine learning models, particularly in situations where minority classes are of critical importance but are underrepresented. SMOTE, by synthesizing new minority class samples, has proven to be instrumental in rectifying such imbalances, thus enhancing classifier performance and reliability [17]. SMOTE operates by generating synthetic instances of the minority class rather than merely duplicating existing instances. It does this by selecting feature space points between existing minority samples and carefully creating new samples that have similar characteristics [18]. This process enriches the dataset and aids classifiers in learning from a more balanced dataset, ultimately leading to better generalization and predictive accuracy. The effectiveness of SMOTE has been demonstrated across various studies, revealing improvements in performance metrics such as F-score and G-mean, even when overall accuracy remains unchanged [17].

One significant advancement in the application of SMOTE is detailed by Hussein et al in their proposal for A-SMOTE, an enhanced preprocessing approach designed specifically for highly imbalanced datasets. By refining the traditional SMOTE method, A-SMOTE aims to further optimize the synthetization process, thus improving accuracy and fostering better predictive outcomes [18]. This variant showcases the versatility of SMOTE principles while introducing new methodologies to tackle specific shortcomings in existing implementations. In the realm of medical applications, the integration of SMOTE has been especially prominent. For instance, Sohn et al employed Multi-Label SMOTE to address class imbalance in a study related to glioblastoma and astrocytomas, highlighting its utility not just in binary classification but also in complex multi-label contexts. This adaptability emphasizes the significance of SMOTE in critical decision-making scenarios where minority class recognition is essential [19], [20].

2.3. Random Forest in Classification

The Random Forest algorithm has gained significant traction in various domains for its robustness and versatility in classification tasks. This ensemble learning method, which constructs multiple decision trees and combines their results, offers a series of advantages that make it particularly effective for dealing with complex datasets across diverse fields. Below is a synthesis of research that highlights the application of Random Forest for classification in various domains. Wang and Wu discuss the application of Random Forest in high-speed network environments, specifically for packet classification at 100 Gbps line rates. Their study illustrates the feasibility of implementing Random Forest as part of a hardware switch pipeline. However, they note that while the potential is promising, the accuracy of packet classification still requires enhancements to meet the critical standards of correctness required in network operations [21], [22]. This reflects Random Forest's adaptability in high-throughput contexts, emphasizing ongoing improvements to bolster its effectiveness.

The work of Schönlau and Zou [23] further elaborates on the robustness of Random Forest in statistical learning applications. They highlight that the improvements in classification performance when using Random Forest over traditional algorithms like logistic regression within social science metrics are often minor [23], [24]. This study underscores the method's stability across diverse applications, reinforcing its utility not just in technical settings but also in social science research. In the field of health and medical research, Muhammad et al. [25] explored the efficacy of Random Forest in predictive models for diabetes mellitus. They utilized the bootstrap aggregation technique inherent to Random Forest to construct multiple decision trees, demonstrating its effectiveness in classification tasks related to health predictions [25]. The integration of such advanced algorithms in healthcare underscores the crucial role of Random Forest in handling complex datasets that influence timely medical decision-making.

In traffic management, Ramdani et al. [26] analyzed vehicular flow classification using Random Forest models. They noted that this supervised learning method is widely adopted due to its efficacy in training data samples and aggregating outputs from multiple trees for enhanced prediction accuracy [26]. This work exemplifies the ongoing reliance on Random Forest in dynamic environments, such as urban traffic systems, where accurate classifications are crucial for traffic management and safety protocols. The educational sector also benefits from the Random Forest algorithm, as highlighted by Pujianto et al. [27]. In this study, Random Forest was deployed to predict the acceptance rates of high school science students based on performance metrics [27]. This innovative application underscores the versatility of Random Forest across various fields, including education.

3. Methodology

3.1. Dataset Overview

The first step in the methodology involves loading and describing the dataset. The Large Cars Dataset is read into a DataFrame using pandas, and basic information about the dataset, such as its shape and data types, is displayed using df.info(). A statistical summary of the numerical features is provided through df.describe(), which gives insight into key characteristics like the mean, standard deviation, and range of each numerical feature. To ensure consistency, the column names are cleaned by removing special characters and spaces using df.columns.str.replace(). This is an important step to avoid issues when working with column names later in the process. The initial rows of the dataset are also printed to give a snapshot of the data, which helps verify that the data has been loaded correctly and is ready for further analysis. This step serves to familiarize the researcher with the dataset and its structure, providing foundational information for the subsequent steps. It also ensures that the dataset is in a clean and usable state before applying any analysis or machine learning techniques. The goal here is to quickly identify the dataset's features and understand the overall structure, including the numerical and categorical data types, which will inform the preprocessing and modeling steps that follow. By exploring this foundational information, the modeler can make informed decisions about how to approach the rest of the pipeline, especially with regard to handling any missing values or outliers in the dataset.

3.2. Exploratory Data Analysis (EDA)

The second phase focuses on EDA, which provides insights into the data and helps identify potential issues such as imbalanced classes. The target variable for this research, VehicleClass, is visualized using a count plot to display its distribution across different classes. This plot helps to identify any imbalances in the target variable, which could affect the performance of the classification model. By visualizing the count of each vehicle class, the researcher can spot if certain classes are underrepresented, which may necessitate the application of techniques like SMOTE later in the pipeline. Additionally, the distribution of the target variable is assessed to ensure that the class labels are appropriately distributed for the task. In addition to visualizing the target variable, a correlation matrix of the numerical features is generated and visualized using a heatmap. This helps uncover any strong relationships between numerical variables, such as engine size and horsepower, which may be predictive of the vehicle class. Understanding these relationships aids in selecting features that are most relevant for classification. The correlation analysis helps identify multicollinearity between variables, allowing for the removal or combination of highly correlated features. If there are insufficient numerical features for correlation analysis, a warning is issued to indicate that this phase may need to be revisited. Overall, EDA is a critical step for gaining initial insights into the data and guiding the preprocessing process.

3.3. Data Preprocessing

Data preprocessing involves preparing the dataset for machine learning by cleaning, transforming, and scaling the features. First, the dataset is divided into features (X) and the target variable (y), where y represents the vehicle class. To enable machine learning algorithms to work with the target variable, LabelEncoder is used to convert the categorical VehicleClass column into numerical labels. This transformation is essential for classification tasks since most machine learning algorithms require numerical input. After encoding the target variable, the categorical features in X are one-hot encoded using pandas.get_dummies(). One-hot encoding transforms categorical variables into binary columns, allowing them to be used in the model. In addition to encoding categorical variables, numerical features are scaled using StandardScaler to standardize their range. Scaling ensures that all features contribute equally to the model, preventing certain features from dominating due to their scale. The dataset is then split into training and testing sets, with an 80/20 split using train_test_split. The stratified splitting ensures that the distribution of the target variable remains consistent in both the training and testing sets. This step is crucial to prevent the model from overfitting to a particular class in the data. Data preprocessing ensures that the dataset is in a format suitable for machine learning, helping improve the performance and accuracy of the model.

3.4. SMOTE Application

SMOTE is applied to the training data to handle class imbalance. Imbalanced class distributions can lead to poor model performance, as the algorithm may become biased toward the majority class. SMOTE generates synthetic samples for the minority classes by creating new, plausible data points based on the existing ones. The number of neighbors used in SMOTE (k_neighbors) is dynamically adjusted based on the size of the smallest class in the training set to ensure that the resampling process is appropriate. If the smallest class has fewer samples than the default k_neighbors, the number is adjusted to prevent over-sampling. After applying SMOTE, the new distribution of classes is visualized to ensure that the class imbalance has been addressed. The sns.countplot() function is used to display the class distribution in the resampled training set. This visualization helps confirm that SMOTE has created a more balanced training set, giving the model a better opportunity to learn from minority classes. By using SMOTE, the training set becomes more representative of all classes, improving the overall classification performance. However, the test set remains untouched by SMOTE to ensure that the evaluation is done on real-world, imbalanced data, providing a realistic measure of model performance.

3.5. Model Selection & Training (Random Forest)

Random Forest is chosen as the model for classification due to its robustness, interpretability, and ability to handle both numerical and categorical features. It is an ensemble method that constructs multiple decision trees and averages their predictions to improve accuracy and reduce overfitting. The model is initialized with 100 estimators, and the training process begins using the resampled data from the SMOTE step. The model is trained on the features and their corresponding encoded target variable, y_train_resampled, using the fit() method. Random Forest is particularly effective for tabular data with a mixture of feature types and is relatively fast compared to more complex models like neural networks. The Random Forest model is trained efficiently by utilizing multiple cores on the machine through the n_jobs=-1 parameter, which speeds up the training process. During training, the model learns to classify vehicles based on the features such as engine size, horsepower, and weight. One of the advantages of Random Forest is that it provides feature importances, which helps in understanding which features contribute most to the classification decision. Once training is complete, the model is ready to be evaluated on the test set. The trained model is capable of making predictions on unseen data, and its performance is assessed through various metrics, including accuracy and F1-score.

3.6. Model Evaluation

After training the model, its performance is evaluated on the test set, which consists of data that was not used during the training process. The accuracy of the model is calculated using the accuracy_score() function, providing a measure of how well the model predicts the correct vehicle class. A classification report is generated using classification_report(), which includes additional metrics such as precision, recall, and F1-score for each class. These metrics provide a more detailed assessment of the model’s ability to distinguish between vehicle classes. The classification report is essential for understanding the model’s strengths and weaknesses, particularly in terms of how it handles each class. A confusion matrix is also generated using confusion_matrix() to visualize the performance of the classifier across all classes. This matrix helps identify specific misclassifications, showing how the model's predictions align with the actual vehicle classes. The confusion matrix is visualized using a heatmap to provide a clearer interpretation of the results. By evaluating the model using these metrics, the researcher can identify areas for improvement, such as class-specific misclassifications or low recall for certain classes. Model evaluation is critical for understanding the practical performance of the classifier and for making informed decisions about potential improvements.

3.7. Model Checkpointing

Once the model has been trained and evaluated, it is saved for future use through model checkpointing. This involves saving the trained Random Forest model, the scaler used for feature scaling, the label encoder, and the processed column order to disk using joblib.dump(). The checkpoint is stored at a predefined path (SAVED_MODEL_PATH) and contains all necessary components to make future predictions or retrain the model without starting from scratch. This approach ensures that the model can be easily reloaded and used for inference on new data without needing to repeat the training and preprocessing steps. Model checkpointing is crucial for deploying the model in production environments or for future experimentation. Saving the model and its components also helps ensure consistency in predictions when the model is used in different environments. The checkpoint file includes the model's parameters, scaling procedure, and encoding information, allowing for seamless integration with new datasets that follow the same preprocessing steps. This step concludes the pipeline, ensuring that the trained model can be reused efficiently and effectively for vehicle classification tasks in the future.

4. Results and Discussion

4.1. Result

4.1.1. Exploratory Data Analysis (EDA)

The dataset consists of 428 entries, with 15 columns representing various features of vehicles. These columns include both numerical and categorical data, such as EngineSize, HorsePower, Weight, and VehicleClass. The statistical summary of numerical features shows that the dataset contains vehicles with engine sizes ranging from 1.3 to 8.3 liters, and horsepower ranging from 73 to 500. The average weight of vehicles is approximately 3,578 lbs, with the lightest vehicle weighing 1,850 lbs and the heaviest 7,190 lbs. The VehicleClass column contains six unique classes: Hybrid, SUV, Sedan, Sports, Truck, and Wagon, with no missing values. The dataset was carefully cleaned, and all columns were prepared for further analysis. The first five rows of the dataset reveal details about various vehicle models and their features, including MSRP, DealerCost, and technical specifications such as EngineSize and HorsePower. The dataset is diverse in terms of vehicle types and includes a range of car classes and regions of origin. This initial exploration sets the foundation for further analysis, enabling the identification of patterns and relationships that will inform the classification model. Understanding the dataset's structure and summary statistics is critical for ensuring that appropriate preprocessing steps are applied and the data is ready for modeling.

EDA helped uncover important insights into the dataset, particularly the distribution of the target variable, VehicleClass. Figure 1 visualize the distribution of vehicle classes before applying any balancing techniques. This plot highlighted imbalances in the dataset, with some classes like SUV and Sedan being overrepresented, while classes like Hybrid and Wagon were underrepresented. These imbalances suggested that class imbalance might affect model performance, motivating the application of SMOTE to address this issue.

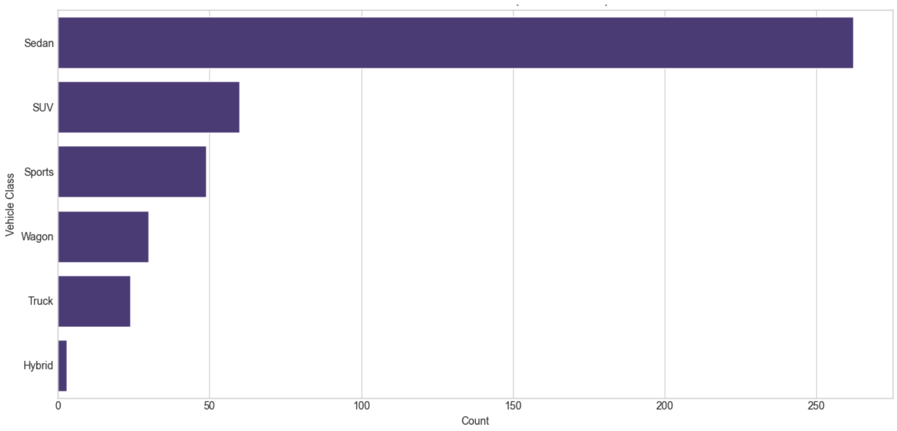

Figure 1. Distribution of Vehicle Classes (Before SMOTE)

Additionally, Figure 2 visualize the relationships between numerical features. The analysis showed strong correlations between EngineSize, HorsePower, and Weight, which suggests that these features are highly influential in classifying vehicle types. Understanding these relationships is key to selecting important features for the model. By conducting EDA, we gained a deeper understanding of the dataset's structure and identified areas that required further attention, such as handling imbalanced data through SMOTE.

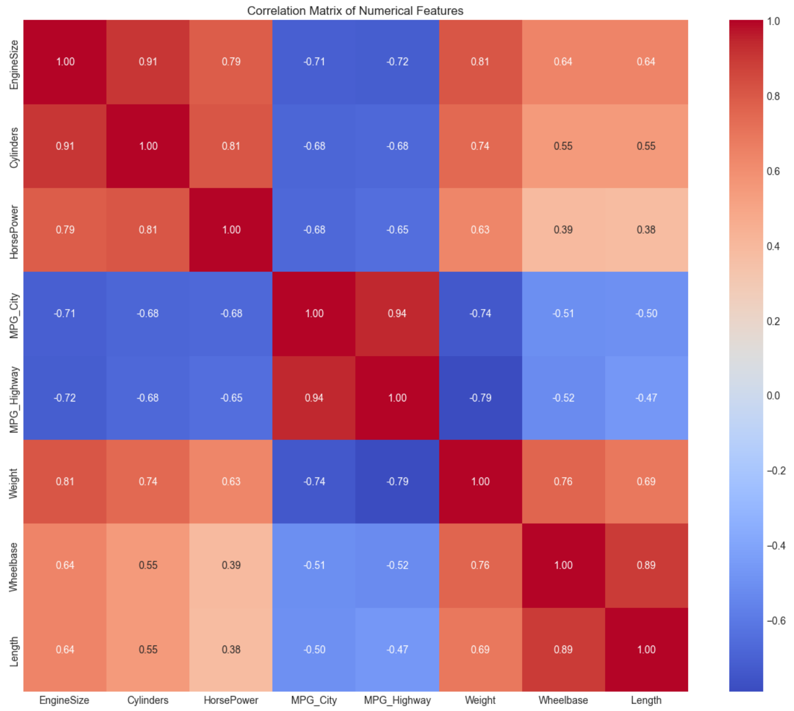

Figure 2. Correlation Matrix of Numerical Features

4.1.2. Results of Data Preprocessing and SMOTE

Data preprocessing involved several steps to prepare the dataset for machine learning. The target variable, VehicleClass, was encoded into numerical labels using LabelEncoder, transforming the class names into integers for classification purposes. The dataset was then split into numerical and categorical features. Categorical features like Brand, Model, and Region were one-hot encoded to create binary columns for each unique category. This transformation allowed the model to work with categorical data effectively. After encoding, the dataset was split into training and testing sets, with 342 samples in the training set and 86 samples in the testing set. The features were scaled using StandardScaler, which normalized the numerical features to have zero mean and unit variance, ensuring that no feature dominated the model due to scale differences. The preprocessing steps were completed successfully, and the resulting dataset was ready for the application of machine learning algorithms. The preprocessed dataset contained 1,306 features after encoding and scaling, making it suitable for training a classification model.

SMOTE was applied to balance the training data by generating synthetic samples for the minority classes. Based on the smallest class size, k_neighbors was set to 2, ensuring that synthetic samples were created by considering the nearest two neighbors. This adjustment helped prevent over-sampling of larger classes, maintaining a more balanced dataset. After applying SMOTE, the training set was expanded from 342 to 1,254 samples, with the class distribution becoming more balanced. The effect of SMOTE was visualized by plotting the class distribution in the resampled training set (Figure 3). The plot showed a more even distribution of classes, which is crucial for training a model that can generalize well across all vehicle types. The use of SMOTE ensured that the model would be exposed to a more representative sample of each class, improving its ability to classify vehicles from underrepresented categories. The successful application of SMOTE helped mitigate the risk of class imbalance affecting the model's performance.

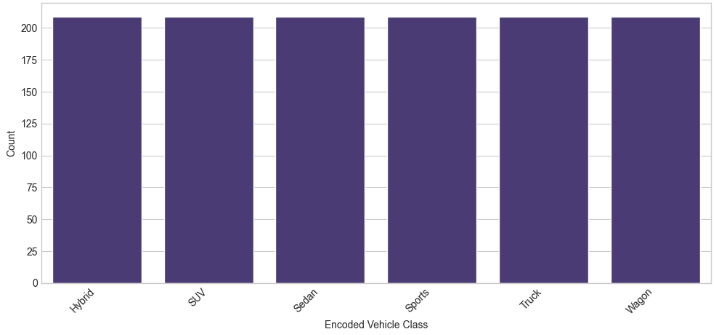

Figure 3. Distribution of Vehicle Classes in Training Data (After SMOTE)

4.1.3. Results of Model Training and Evaluation

The Random Forest classifier was chosen for this classification task due to its robustness and ability to handle both numerical and categorical data. The model was trained using the resampled training set, which consisted of 1,254 samples. The Random Forest model was initialized with 100 trees (n_estimators=100) and trained using multiple cores (n_jobs=-1) to speed up the process. The model was successfully trained on the resampled data, and the training process was completed without any issues. Random Forest's ability to handle high-dimensional data made it an ideal choice for this task, as the dataset contained a large number of features after one-hot encoding. The model's performance was expected to improve with the diverse set of decision trees, each trained on different subsets of the data. After training, the model was ready for evaluation, and its performance was assessed using the testing set to ensure that it could generalize well to unseen data.

The model was evaluated on the test set, achieving an accuracy of 77.91%. The classification report provided detailed metrics, including precision, recall, and F1-score for each vehicle class. Notably, the model performed well in classifying Sedan vehicles, achieving a recall of 1.00 and an F1-score of 0.88. However, the model struggled with the Hybrid and Wagon classes, where it achieved precision and recall of 0.00, indicating that these classes were poorly predicted. The confusion matrix further highlighted these issues, showing that certain classes were frequently misclassified. Despite the model's strong performance with some vehicle classes, the overall results were mixed, with a macro average recall of 0.42, suggesting that the model could be improved, particularly for underrepresented classes. The confusion matrix, which was visualized using a heatmap (Figure 4), provided further insight into the misclassifications and helped pinpoint areas where the model's performance could be enhanced. The evaluation metrics suggest that while the Random Forest model performed reasonably well overall, there is room for improvement, especially for certain vehicle types.

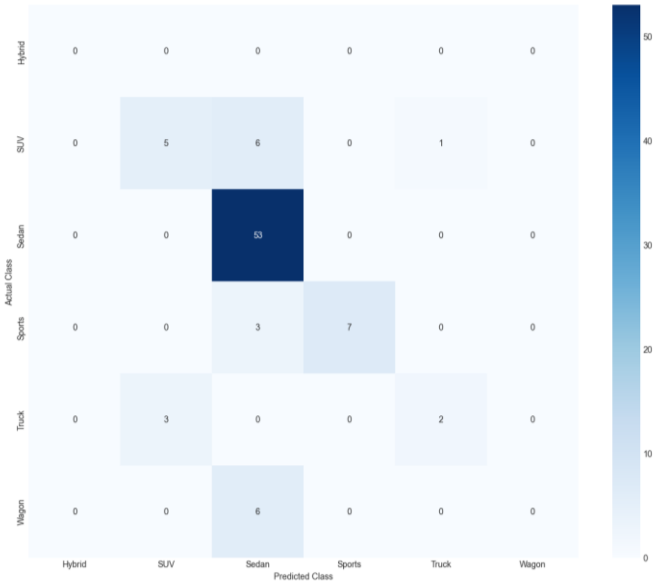

Figure 4. Confusion Matrix

After successfully training and evaluating the model, the final Random Forest model, along with the label encoder, scaler, and feature list, was saved to a checkpoint file using joblib. This saved model, stored at the path 'random_forest_vehicle_classifier.joblib', allows for future use in vehicle classification tasks without the need to retrain the model. The checkpoint includes all the necessary components to preprocess new data, make predictions, and ensure consistency across different environments. Saving the trained model and preprocessing components ensures that the model can be easily deployed or used for further experimentation. The model checkpoint allows for efficient reuse in real-world applications, such as vehicle classification in online platforms or automotive industry analysis. This final step concludes the research pipeline, providing a robust and reusable model for classifying vehicles based on their technical specifications.

4.2. Discussion

The results of the classification model reveal several important insights into the dataset and the vehicle classification task. One of the most striking observations is the strong performance of the Random Forest model in classifying Sedan vehicles, with perfect recall (1.00) and a high F1-score (0.88). This suggests that the features, such as engine size, horsepower, and weight, are highly indicative of the Sedan class, making it easier for the model to correctly identify this vehicle type. However, this success is not consistent across all vehicle classes, with other categories such as Hybrid and Wagon showing poor performance, as evidenced by a precision and recall of 0.00 for these classes. These discrepancies highlight the challenges of dealing with class imbalances, even after applying SMOTE, and suggest that more sophisticated resampling or feature engineering techniques may be needed to improve performance for minority classes.

The class imbalance in the original dataset was a significant challenge, and while SMOTE helped balance the training data, the model still struggled with underrepresented classes. For example, the Hybrid and Wagon classes had very few instances in the training set, which led to poor model performance despite the resampling technique. The confusion matrix further emphasized this issue, showing that the model frequently misclassified these minority classes as other vehicle types. This suggests that although SMOTE improves balance, it may not always be sufficient in cases where the class distribution is extremely skewed. Future work may explore alternative strategies such as class-weight adjustments, more aggressive data augmentation, or the use of more complex algorithms that are better suited to handle imbalanced datasets.

In terms of feature importance, the model likely relied heavily on attributes like HorsePower, EngineSize, and Weight, which showed strong correlations in the EDA phase. These features are intuitively relevant to vehicle classification, especially for distinguishing between larger, more powerful vehicle types like SUVs and Trucks, and smaller, more fuel-efficient classes like Sedans and Hybrids. However, the relatively low performance for certain vehicle classes suggests that additional features, such as more granular information about the vehicle's design or performance characteristics, could improve the model's ability to distinguish between difficult-to-classify vehicles. This highlights the need for more comprehensive data and potentially feature engineering to better capture the nuances of different vehicle types and improve the overall classification accuracy.

5. Conclusion

In this study, the Random Forest classifier was used to classify vehicles based on their technical specifications. The key findings reveal that the model performed well in identifying `Sedan` vehicles, achieving a perfect recall and high F1-score. However, the model struggled with underrepresented classes such as `Hybrid` and `Wagon`, highlighting the challenge of class imbalance, even after applying SMOTE. This indicates that while SMOTE improved the balance of the training set, additional techniques may be required to further enhance the classification of minority classes. Overall, the study provided insights into how certain features, such as engine size and horsepower, are critical for vehicle classification, but also pointed to the limitations of relying on a single machine learning approach for a highly imbalanced dataset. This research contributes to vehicle classification research by demonstrating the application of Random Forest in a real-world scenario and evaluating its performance in a diverse dataset. It offers insights into the importance of feature selection and the challenges posed by class imbalances in predictive modeling. However, the approach has limitations, including potential biases in the dataset, such as unequal representation of vehicle types. For future work, testing with other algorithms like XGBoost or deep learning methods, as well as incorporating larger, more diverse datasets, could improve model performance. The practical implications of this study are significant for vehicle market segmentation, enabling manufacturers and dealerships to better understand vehicle categories, optimize their inventory, and improve targeted marketing strategies based on accurate vehicle classification models.

6. Declarations

6.1. Author Contributions

Conceptualization: D.S, T.W.; Methodology: D.S, T.W.; Software: D.S.; Validation: T.W.; Formal Analysis: D.S.; Investigation: D.S.; Resources: T.W.; Data Curation: D.S.; Writing – Original Draft Preparation: D.S.; Writing – Review and Editing: D.S, T.W.; Visualization: D.S.; All authors have read and agreed to the published version of the manuscript.

6.2. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

6.3. Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

6.4. Institutional Review Board Statement

Not applicable.

6.5. Informed Consent Statement

Not applicable.

6.6. Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] V. Velisavljević, E. Cano, V. Dyo, and B. Allen, “Wireless Magnetic Sensor Network for Road Traffic Monitoring and Vehicle Classification,” *Transp. Telecommun. J.*, vol. 17, no. 4, pp. 274–288, 2016., doi: 10.1515/ttj-2016-0024.

[2] I. Atouf, W. Y. Al Okaishi, A. Zaaran, I. Slimani, and M. Benrabh, “A Real-Time System for Vehicle Detection with Shadow Removal and Vehicle Classification Based on Vehicle Features at Urban Roads,” *Int. J. Power Electron. Drive Syst. (IJPEDS)*, vol. 11, no. 4, pp. 2091–2098, 2020, doi: 10.11591/ijpeds.v11.i4.pp2091-2098.

[3] H. Nambo, I. Tahyudin, T. Nakano, and T. Yamada, “Comparison of Deep Learing Algorithms for Indoor Monitoring using Bioelectric Potential of Living Plants,” *in Proc. 3rd Int. Conf. Inf. Technol., Inf. Syst. Electr. Eng. (ICITISEE)*, Yogyakarta, Indonesia, Nov. 2018, pp. 110–113, doi: 10.1109/ICITISEE.2018.8720992.

[4] S. Rajab, M. O. Al Kalaa, and H. H. Refai, “Classification and Speed Estimation of Vehicles via Tire Detection using Single-Element Piezoelectric Sensor,” *J. Adv. Transp.*, vol. 50, no. 7, pp. 1366–1385, 2016, doi: 10.1002/atr.1406.

[5] M.-T. Lai and T. Hariguna, “Analyzing Company Hiring Patterns Using K-Means Clustering and Association Rule Mining: A Data-Driven Approach to Understanding Recruitment Trends in the Digital Economy,” *J. Digit. Soc.*, vol. 1, no. 1, pp. 20-43, 2025, doi: 10.63913/jds.v1i1.2.

[6] Z. Wei, W. Guo, and X. Zheng, “Enhancing Small Sample Vehicle Type Classification Accuracy Through Background-Free Data Augmentation,” *in Proc. SPIE 13078, Eleventh Int. Conf. on Machine Vision (ICMV 2024)*, Amsterdam, The Netherlands, Jan. 2025, vol. 13078, pp. 1307811–1307817, doi: 10.1117/12.3061655.

[7] S. Noh and M. Y. Jeon, “Vehicle Detection Using Local Size-Specific Classifiers,” *IEICE Trans. Inf. Syst.*, vol. E99-D, no. 12, pp. 3034–3037, 2016, doi: 10.1587/transinf.2016EDP7101.

[8] H. Abedi, S. Luo, V. Mazumdar, M. M. Y. R. Riad, and G. Shaker, “AI-Powered In-Vehicle Passenger Monitoring Using Low-Cost mm-Wave Radar,” *IEEE Access*, vol. 10, pp. 12601–12612, 2022, doi: 10.1109/ACCESS.2021.3138051.

[9] J. Li, Y. Sun, J. Ren, Y. Wu, and Z. He, “Machine Learning for in-Hospital Mortality Prediction in Critically Ill Patients with Acute Heart Failure: A Retrospective Analysis Based on MIMIC -IV Databases,” *Res. Sq. Prepr.*, 2024, doi: 10.21203/rs.3.rs-3834698/v1.

[10] H. T. Sukmana and L. K. Oh, “Evaluating the Impact of Affirmative Action on Student Selection Outcomes: A Data Mining Approach Using SKD Test Performance at STMKG,” *Artif. Intell. Learn.*, vol. 1, no. 1, pp. 54-74, 2025, doi: 10.63913/jcl.v1i2.7.

[11] A. Kaya, A. S. Keçeli, C. Catal, and B. Tekinerdoğan, “The Impact of Feature Types, Classifiers, and Data Balancing Techniques on Software Vulnerability Prediction Models,” *J. Softw. Evol. Process.*, vol. 31, no. 11, pp. e2164, 2019, doi: 10.1002/smr.2164.

[12] X. Zhou, J. Tian, and M. Su, “Tour-Route-Recommendation Algorithm Based on the Improved AGNES Spatial Clustering and Space-Time Deduction Model,” *ISPRS Int. J. Geo-Inf.*, vol. 11, no. 2, p. 118, 2022, doi: 10.3390/ijgi11020118.

[13] S. F. Pratama and A. M. Wahid, “Fraudulent Transaction Detection in Online Systems Using Random Forest and Gradient Boosting,” *J. Cyber Law*, vol. 1, no. 1, pp. 88-115, 2025.

[14] N. A. Ahmad, E. Sylviana binti M. Zahid, I. B. A. Halim, A. Termizi bin Ab Lateh, and G. Ghutai, “Financial Literacy Among Investors,” *Int. J. Acad. Res. Bus. Soc. Sci.*, vol. 12, no. 4, pp. 1411–1426, 2022, doi: 10.6007/ijarbss/v12-i4/13120.

[15] R. Savitha, K. Y. Chan, P. P. San, S. H. Ling, and S. Suresh, “A Hybrid Deep Boltzmann Functional Link Network for Classification Problems,” *in Proc. IEEE Symp. Ser. Comput. Intell. (SSCI)*, Athens, Greece, Dec. 2016, pp. 1–8, doi: 10.1109/SSCI.2016.7850114.

[16] D. Kleyko, R. Hostettler, N. Lyamin, W. Birk, U. Wiklund, and E. Osipov, “Vehicle Classification Using Road Side Sensors and Feature-Free Data Smashing Approach,” *in Proc. IEEE 19th Int. Conf. Intell. Transp. Syst. (ITSC)*, Rio de Janeiro, Brazil, Nov. 2016, pp. 1232–1237, doi: 10.1109/ITSC.2016.7795877.

[17] G. Douzas, F. Bação, J. Fonseca, and M. Khudinyan, “Imbalanced Learning in Land Cover Classification: Improving Minority Classes’ Prediction Accuracy Using the Geometric SMOTE Algorithm,” *Remote Sens.*, vol. 11, no. 24, pp. 3040, 2019, doi: 10.3390/rs11243040.

[18] A. S. Hussein, T. Li, C. W. Yohannese, and K. Bashir, “A-Smote: A New Preprocessing Approach for Highly Imbalanced Datasets by Improving SMOTE,” *Int. J. Comput. Intell. Syst.*, vol. 12, no. 2, pp. 1412–1422, 2019, doi: 10.2991/ijcis.d.191114.002.

[19] B. Sohn, Y-C. Ryu, H. Kim, J. Park, M. Kim, J. H. Chang, S-K. Lee, and S-H. Park, “Radiomics-Based Prediction of Multiple Gene Alteration Incorporating Mutual Genetic Information in Glioblastoma and Grade 4 Astrocytoma, IDH-mutant,” *J. Neuro-Oncol.*, vol. 155, no. 2, pp. 141–150, 2021, doi: 10.1007/s11060-021-03870-z.

[20] A. R. Yadulla, M. H. Maturi, K. Meduri, and G. S. Nadella, “Sales Trends and Price Determinants in the Virtual Property Market: Insights from Blockchain-Based Platforms,” *Int. J. Res. Metaverse*, vol. 1, no. 2, pp. 113–126, 2024, doi: 10.47738/ijrm.v1i2.9.

[21] M. Wang, A. Dev, and Q. Zhou, “Do Cryptocurrencies Exhibit Herding Behavior? Evidence From CSSD and CSAD Approaches,” *J. Student Res.*, vol. 12, no. 4, pp. 1–8, 2023, doi: 10.47611/jsrhs.v12i4.5649.

[22] D. Wu, B. Sun, and M. Shang, “Hyperparameter Learning for Deep Learning-Based Recommender Systems,” *IEEE Trans. Serv. Comput.*, vol. 16, no. 4, pp. 2699–2712, 2023, doi: 10.1109/TSC.2023.3234623.

[23] M. Schonlau and R. Y. Zou, “The Random Forest Algorithm for Statistical Learning,” *Stata J. Promot. Commun. Stat. Stata*, vol. 20, no. 1, pp. 3–29, 2020, doi: 10.1177/1536867X20909688.

[24] Berlilana and A. M. Wahid, “Time Series Analysis of Bitcoin Prices Using ARIMA and LSTM for Trend Prediction,” *J. Digit. Mark. Digit. Currency*, vol. 1, no. 1, pp. 84-102, 2024, doi: 10.47738/jdmdc.v1i1.1.

[25] L. J. Muhammad, E. A. Algehyne, and S. S. Usman, “Predictive Supervised Machine Learning Models for Diabetes Mellitus,” *SN Comput. Sci.*, vol. 1, no. 5, pp. 1–7, 2020, doi: 10.1007/s42979-020-00250-8.

[26] N. Ramdani, S. S. Prasetyowati, and Y. Sibaroni, “Performance Analysis of Bandung City Traffic Flow Classification With Machine Learning and Kriging Interpolation,” *Build. Inf. Technol. Sci. (BITS)*, vol. 4, no. 2, pp. 694–704, 2022, doi: 10.47065/bits.v4i2.1972.

[27] U. Pujianto, A. R. Taufani, and J. A. Aziz, “Random Forest Algorithm for Algorithm for Prediction of High School Science Students Acceptance SNMPTN Based on Students Assesment Report,” *J. Inform.*, vol. 15, no. 3, pp. 29–37, 2021, doi: 10.26555/jifo.v15i3.a25413.