FlockSense: Data-Driven Model for Autonomous Traffic Management and Navigation

Adiba Sharif

October 2023

Abstract

Traffic management and navigation for autonomous vehicles pose significant challenges in the era of smart transportation. Efficient traffic management is crucial to reduce congestion, improve fuel efficiency, and enhance passenger safety. This work presents a novel approach to address this problem using machine learning techniques.The model leverages the Swarm Behaviour Classification dataset from UCI to train various machine learning models. The methodology involves hyperparameter tuning methods for model optimization and ensemble learning strategies for improving model performance. The application of these techniques has resulted in improved training accuracy and similar accuracies with slight variations after applying different hyperparameter tuning techniques and ensemble learning methods.he best result was achieved by applying a voting classifier (pasting) on the Support Vector Machines (SVM) model after hyperparameter tuning. The hyperparameter tuning was performed using GridSearchCV,Randomized SearchCV , Optuna - Automate Hyperparameter Tuning, Sequential Model-Based Optimization ,Genetic Algorithms (TPOT Classifier), and Bayesian Optimization - Automate Hyperparameter Tuning (Hyperopt). After applying various voting classifiers, the best performed model achieved a training accuracy of 0.955075 and a test accuracy of 0.953419.This work contributes to the quest for more efficient and safer traffic management and navigation systems for autonomous vehicles. The results indicate that this approach has potential for further research and development, offering a promising solution to the challenges of traffic management and navigation for autonomous vehicles. The novelty of this work lies in its unique application of machine learning techniques to a unique dataset, providing a new perspective in the field of autonomous vehicles.

Keywords

Traffic management and navigation for autonomous vehicles, Machine learning, Voting algorithm, Hyperparameter tuning, Cross-Validation

1 Introduction

In the natural world, we often see complex systems exhibiting remarkably coordinated behavior. A prime example of this is a flock of birds moving seamlessly in the sky, their movements synchronized to prevent any clashes. This phenomenon, known as swarm behavior, is not only fascinating but also holds potential solutions to some of our most pressing problems.One such problem is traffic management. Traffic management and autonomous vehicles are rapidly evolving fields that promise to revolutionize the way we travel. Autonomous Vehicles (AVs) have the potential to disrupt traditional transportation systems, requiring an uninterrupted, continuous stream of data and information based on complex traffic datasets and predictive measurements [1, 2]. However, the integration of AVs into our existing traffic systems presents unique challenges [1, 3].

The increasing urbanization and population growth have led to escalating traffic congestion problems worldwide [4, 5]. Inefficient traffic management not only leads to longer travel times but also contributes to environmental pollution and decreased quality of life [4].

This study aims to take the first step towards understanding the primary causes of traffic congestion in urban areas and to build a model for efficient traffic management and autonomous vehicles. The model is built using the Swarm Behaviour Classification dataset available on UCI [7, 6]. The dataset provides a rich source of information on swarm behaviour, which can be instrumental in understanding and predicting traffic patterns.

Machine learning techniques have been utilized to tackle this issue [1, 8]. These techniques can help in predicting traffic patterns, optimizing traffic flow during congestion periods, and making infrastructure intelligent enough to handle traffic congestion and balance traffic flow efficiently [1, 9].

The model building process involves several steps:

1. Comprehensive preprocessing of the Swarm Behaviour Classification dataset, including outlier detection and feature scaling.

2. Evaluation of various machine learning algorithms on the dataset .

3. Hyperparameter tuning of the SVM model to improve performance .

4. Application of feature selection methods to reduce dimensionality and improve model performance.

5. Implementation of ensemble learning using a voting classifier on the SVM model to enhance predictive performance .

6. Export of the best-performed model for future predictions. This model was obtained after applying the Pasting method in ensemble learning .

This model introduces a unique approach to traffic management and autonomous vehicles. It's the first of its kind to integrate a comprehensive preprocessing of the dataset, evaluation of various machine learning algorithms, hyperparameter tuning, feature selection, and ensemble learning . The model's performance has been optimized using advanced techniques, setting it apart from existing models in the field . The model is also distinctive in its use of the Swarm Behaviour Classification dataset, marking a novel application of this data in the realm of traffic management and autonomous vehicles . This work represents a significant advancement in the development of intelligent traffic management systems and autonomous vehicles .

The article follows a structured sequence: Section 2 delves into related works, Section 3 outlines the methodology employed in this study, and Section 4 scrutinizes the obtained results. Section 5 elaborates on how these results align with our research objectives. Section 6 delineates the limitations of this study, and finally, Section 7 encompasses the conclusion, including insights on our future endeavors.

2 Literature Review

The increasing complexity of traffic systems and rapid advancement of autonomous vehicle technology have necessitated efficient traffic management [10, 11, 12]. The potential to improve safety, efficiency, and sustainability in transportation has intrigued researchers to explore innovative solutions [13, 14]. This has led to a surge in studies focusing on the application of machine learning and data-driven methods in this field [10, 12, 13, 14, 11].

2.1 Traffic Management and Autonomous Vehicles Using Machine Learning

There has been significant work in the field of traffic management and autonomous vehicles using machine learning. For instance, a study [10] proposed a traffic management system for autonomous vehicles using policy-based Deep Reinforcement Learning (DRL) and intelligent routing. The system dynamically adjusts traffic signals according to the current congestion situation on intersections [10]. Another research [11] provided a comprehensive review on intelligent traffic management using machine learning algorithms. It discussed the use of artificial intelligence and machine learning to provide optimal solutions for varying traffic congestion [11]. A study [10] developed a novel machine learning enabled traffic prediction method and integrated it with a speed optimization algorithm for connected and autonomous electric vehicles. The traffic prediction was based on a hybrid macroscopic traffic flow model [10]. Then, [11] studied the decision-making problem of autonomous vehicles in traffic. The interaction between an autonomous vehicle and the environment was modeled as a stochastic Markov decision process (MDP), considering the driving style of an experienced driver as the target to be learned [11]. A study [15]provided a comprehensive review on intelligent traffic management using machine learning algorithms. It discussed the use of artificial intelligence and machine learning to provide optimal solutions for varying traffic congestion [15]. Another study developed a novel machine learning enabled traffic prediction method and integrated it with a speed optimization algorithm for connected and autonomous electric vehicles. The traffic prediction was based on a hybrid macroscopic traffic flow model [16].

2.2 Studies using Swarm Behaviour Classification Dataset

The Swarm Behaviour Classification dataset has been used in various studies. A research [12] applied machine learning to represent a swarm's collective movements. It used various machine learning algorithms to classify the swarm behavior in terms of flocking, aligned and grouped, or non-flocking, non-aligned, and non-grouped [12]. Another study [13] applied a purely data-driven method to learn local interactions of homogeneous swarms through observation data and to generate similar swarming behaviour using the learned model [13]. A research [12] applied a purely data-driven method to learn local interactions of homogeneous swarms through observation data and to generate similar swarming behaviour using the learned model [12].The study [13] focused on the autonomous recognition of swarm behaviour in Unmanned Ground Vehicles (UGVs) using transfer learning [13]. The study A study [17] proposed a transfer learning approach for recognising swarming in simulated and real robots.Results show that this value function can detect swarming in at least 89% of cases [17]. Another study [18] presented a technical review related to the swarm robotics tasks, categorizing them into low-level and high-level tasks .The discussion scopes of the low-level tasks include definition and purpose, classification and methods and High-level tasks are discussed in terms of related skills and methods [18].

2.3 Comparative Analysis of Related Works

Here is a Comparative Analysis of Related Works based on the information available:

Work	Focus	Methodology	Advantage	Limitation
[10]	Traffic management for autonomous vehicles	Policy-based DRL and intelligent routing	Dynamically adjusts traffic signals according to the current congestion situation on intersections	It does not consider real-time traffic data, which our model incorporates for dynamic adjustments.
[15]	Intelligent traffic management	Various machine learning algorithms	Provides optimal solutions for varying traffic congestion	It does not implement a robust feature selection method, which we have incorporated in our model.
[16]	Traffic management for autonomous vehicles	Hybrid macroscopic traffic flow model	Developed a traffic prediction method and integrated it with a speed optimization algorithm	It lacks traffic prediction, a crucial aspect of our model.
[11]	Decision making for autonomous vehicles	Stochastic Markov decision process (MDP)	Modeled the interaction between an autonomous vehicle and the environment	It lacks a comprehensive machine learning approach, which we have implemented in our model.
[12]	Swarm behaviour classification	Various machine learning algorithms	Applied machine learning to represent a swarm's collective movements	It does not use swarm behavior data, a key component of our model.
[13]	Swarm behaviour classification	Data-driven method	Learned local interactions of homogeneous swarms through observation data	It does not address varying road network capacities, which our model takes into account.
[17]	Swarm behaviour classification	Transfer learning	Focused on the autonomous recognition of swarm behaviour in UGVs	It does not address traffic management in the context of autonomous vehicles, a key focus of our model.
[18]	Swarm behaviour classification	Various methods	Discussed various swarm robotics behaviours and tasks	It does not consider the impact of autonomous vehicles on traffic management, a key focus of our model.

In conclusion, after the comparative analysis of related works reveals several limitations in existing studies. Notably, i) there is a lack of attention to class imbalance in the dataset, a critical factor in model performance. ii) Additionally, most works have not utilized diverse hyperparameter tuning methods, which can significantly optimize model performance. iii)The application of advanced feature selection is also largely absent, which can improve computational efficiency and model accuracy. Iv) Lastly, the enhancement of predictive accuracy through ensemble learning is not commonly observed. These limitations have been addressed in our work, contributing to the advancement of traffic management systems and autonomous vehicles.

3 Methodology

The following diagram represents the methodology of our project on traffic management and autonomous vehicles using the Swarm Behaviour Classification dataset. It begins with data cleaning and pre-processing, which includes outlier detection using Z-score, handling imbalanced data using SMOTE, and feature scaling using the Min-Max Scaler. The processed data is then split into training and test sets. A classification algorithm is applied to the training data, leading to the construction of the classification model. The model incorporates feature selection, hyper-parameter tuning, and ensemble learning using a voting classifier, ensuring a robust and accurate prediction of swarm behaviour.

3.1 Data Collection

The model was developed using the Swarm Behaviour dataset from UCI, which includes 23,309 instances with 2,401 attributes such as xm, ym (position), xVeln, yVeln (velocity), xAm, yAm (alignment), xSm, ySm (separation), xCm, yCm (cohesion) of each boid, and nACm, nSm (number of boids in the radius of Alignment/Cohesion and Separation). The data which contains multivariate data on swarm behavior, including position, velocity, alignment, separation, and cohesion vectors of each boid [6, 7], split into a 70:30 ratio for training and testing, provides a comprehensive view of swarm behavior, facilitating robust model evaluation in real or simulated environments .

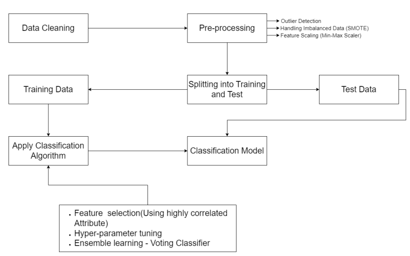

Figure 1: Methodology of the work

3.2 Data pre-processing

In the study, the initial Swarm Behaviour dataset, swarm_df, comprising 23,309 entries across 2,401 columns, was examined. The target variable, 'Swarm Behaviour', had an imbalanced distribution with 15,355 entries for class 0 and

7,954 for class 1. Outliers were identified and eliminated using the Z-Score method, resulting in a filtered dataset. Subsequently, the data was partitioned into training and testing sets following a 70:30 split. Finally, feature scaling was implemented using the min-max scaler to normalize the range of feature values, ensuring equal contribution from all features to the model's performance. This rigorous pre-processing pipeline facilitated the creation of a robust and reliable model.

3.3 Handling Skewed Dataset

In addressing the skewed distribution of the 'Swarm Behaviour' target variable, stratification was initially applied with the KNN algorithm on both scaled and unscaled data. This yielded a train accuracy of 0.962 on unscaled data and 0.963 on scaled data, and a test accuracy of 0.943192 on unscaled data and 0.940432 on scaled data. To further improve the model, the Synthetic Minority Over-sampling Technique (SMOTE) was implemented, equalizing the class distribution to 7,973 entries each for class 0 and class 1. The data was then partitioned into training and testing sets using x_smote and y_smote, resulting in a train accuracy of 0.966 and a test accuracy of 0.944760. Consequently, SMOTE proved more effective than stratification in handling the skewed dataset.

3.4 Model Building

In the model building phase, an array of classifiers was utilized to develop a predictive model. These classifiers included Logistic Regression, Decision Trees, Random Forest, Support Vector Machines (SVM), k-Nearest Neighbors (kNN), Gradient Boosting Algorithms (XGBoost), Linear Discriminant Analysis (LDA), and Quadratic Discriminant Analysis (QDA). Each of these classifiers was trained on the pre-processed dataset and evaluated based on their training and testing accuracy. To further enhance the predictive power of the individual models, ensemble learning was implemented. A soft voting approach was used in the ensemble method, which predicts the final class label based on the argmax of the sums of the predicted probabilities. Among all, the SVM classifier demonstrated superior performance, achieving a train accuracy of 0.960 and a test accuracy of 0.948044, thereby proving to be the most effective model for this dataset. This indicates that the SVM classifier was able to capture the underlying patterns in the dataset most effectively, making it the most suitable model for this particular task.

3.5 ANN (Artificial Neural Network) Experiment

In addition to the aforementioned classifiers, an experiment was conducted with an Artificial Neural Network (ANN). ANNs are powerful computational models capable of capturing complex patterns in data. In this case, the ANN was trained on the same pre-processed dataset and its performance was evaluated based on accuracy. Remarkably, the ANN achieved an accuracy of 0.9540, demonstrating its effectiveness in modeling swarm behavior. This experiment underscores the potential of neural networks in enhancing the predictive performance of the model.

3.6 Feature Selection & cross-validation

In the feature selection phase, a correlation matrix was utilized to identify the top 248 attributes with the highest correlation to the target variable. Two methods were employed for feature selection on the SVM model: Pearson Correlation and the SelectKBest method. Pearson Correlation measures the linear relationship between two datasets [27], while the SelectKBest method selects features according to the k highest scores of a specified scoring function [28]. Upon comparison, the model's train and test accuracies were found to be the same after using these two methods of feature selection. The final decision on the feature selection method was made after implementing the Cross Validation using K fold cross validation technique,indicating its equal performance in selecting the most relevant features for the SVM model.

3.7 HyperParameters Optimization

Post feature selection using the Pearson Correlation and SelectKBest method, hyperparameter tuning was performed on the SVM model. A range of techniques were employed for this purpose, including Manual Hyperparameter Tuning, GridSearchCV, Randomized SearchCV, Sequential Model Based Optimization, Optuna- Automate Hyperparameter Tuning, Genetic Algorithms (TPOT Classifier), and Bayesian Optimization -Automate Hyperparameter Tuning (Hyperopt). Among these, the SVM model demonstrated similar accuracies with slight variations after applying different hyperparameter tuning techniques. The best accuracy achieved was 0.955075 and 0.953419 on the train and test data , indicating a robust and well-tuned model.

3.8 Ensemble Learning - Voting Classifier

In the final phase of model optimization, ensemble learning was applied to the SVM model using a voting classifier. Techniques such as Bagging, Pasting, and Out of Bag Evaluation were implemented. Interestingly, the accuracy achieved after applying the voting classifier on all the hyperparameter tuning models was almost identical to that obtained from Pasting . This consistency in performance led to the selection of the model obtained from Pasting as the best-performed model, which was subsequently exported. The highest accuracy achieved in this phase was 0.955075 and 0.953419 on the train and test data , demonstrating the effectiveness of ensemble learning in enhancing the model's predictive performance.

Therefore, the best performed model is the SVM model with a voting classifier using the pasting technique after hyperparameter tuning .

4 Results and Analysis

In this study, various models were built to find the most appropriate and autonomous model for the refinement of the model. In this study, precision, recall, and F1 score are used to measure the performance of the models regarding the refinement of the model.

The accuracy of a model is defined as the ratio of correctly predicted instances to the total instances in the dataset. It can be calculated using the following formula: [20]

$$\text{Accuracy} = \frac{\text{True Positives (TP)} + \text{True Negatives (TN)}}{\text{Total Instances}}$$

Precision is the ratio of correctly predicted positive instances to the total predicted positive instances. The formula for precision is: [19]

$$\text{Precision} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positives (FP)}}$$

Recall (Sensitivity) - the ratio of correctly predicted positive instances to the total actual positive instances. The formula for recall is: [19]

$$\text{Recall} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negatives (FN)}}$$

F1 Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account. The formula for the F1 score is: [19]

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

	Predicted: Yes	Predicted: No
Actual: Yes	True Positives (TP)	False Negatives (FN)
Actual: No	False Positives (FP)	True Negatives (TN)

Table 1: Confusion Matrix

[21]

Method	**Train Accuracy**	**Test Accuracy**
Stratify (unscaled data)	0.962	0.943192
Stratify (scaled data)	0.963	0.940432
SMOTE	0.966	0.944760

Table 2: Data Pre-processing and Feature Selection

Class	Precision	Recall	F1-Score	Support
1.0	0.50	1.00	0.66	1663
0.0	0.00	0.00	0.00	1686
Accuracy	-	-	0.50	3349
Macro Avg	0.25	0.50	0.33	3349
Weighted Avg	0.25	0.50	0.33	3349

Table 3: ZeroR & Custom ZeroR Classifier

Classifier	Train Accuracy	Test Accuracy	Precision	Recall	F1-Score	Ensemble Test Accuracy Difference
Logistic Regression	0.979	0.939385	0.94	0.94	0.94	0.0
Decision Tree Classifier	0.990	0.941176	0.94	0.94	0.94	0.000299
Random Forest Classifier	0.990	0.940579	0.94	0.94	0.94	0.000597
SVC	0.960	0.948044	0.95	0.95	0.95	0.0
KNeighbors Classifier	0.969	0.946551	0.95	0.95	0.95	0.0
XGB Classifier	0.990	0.942371	0.94	0.94	0.94	0.0
Linear Discriminant Analysis	0.981	0.940281	0.94	0.94	0.94	0.0
Quadratic Discriminant Analysis	0.988	0.942669	0.94	0.94	0.94	0.0
ANN	-	0.9540	0.96	0.95	0.95	0.0

Table 4: Accuracy Using Various Classifiers before and after implementing ensamble learning (before hyperparameter-tuning)

Here, in Table 2 & 3 shows the initial steps involved data pre-processing to handle skewness using SMOTE , showing substantial improvements with 0.944760 of test accuracy over stratification. And, baseline models (ZeroR & Custom ZeroR) presented low accuracy, indicating the complexity of the classification task.

4.1 Performance of models using various classifiers

Upon analysis from Table 4, the Support Vector Machines (SVM) demonstrated the highest accuracy of 0.960 and 0.948044 on train and test data both before and after ensemble learning. This stability highlights SVM's robustness in classification tasks. Even though models like k-Nearest Neighbors (kNN), XGBoost (Xg) and Quadratic Discriminant Analysis (QDA) remained consistent without notable improvements through ensemble techniques ; these models maintained competitive test accuracies with kNN holding at 94.65%,Xg at 94.23% and QDA at 94.26%. Notably, only Decision Trees (dt) and Random Forest Classifier(rf) show a minor changes of 0.000299 and 0.000597 from post-ensemble learning. From Table 4, we can notice that the Artificial Neural Network (ANN) model was configured with specific hyperparameters for optimal performance and exhibited a notable accuracy rate of 95.40%. This accuracy reflects its capability to make precise predictions and classifications with an impressive level of accuracy.

Method	Train Accuracy	Test Accuracy
Pearson Correlation	0.956099	0.952822
SelectKBest	0.956099	0.952822

Table 5: Feature Selection on SVM

Method	Average Cross-Validation Score
Pearson Correlation	0.9533241199537084
SelectKBest	0.9533241199537084

Table 6: Cross Validation Using K Fold Cross Validation Technique

4.2 Feature Selection Techniques and Cross-Validation

In Table 5 & 6 shows the feature selection was performed using the Pearson Correlation & SelectKBest methods, reducing the feature set to the top 248 highly correlated attributes. This step helped in minimizing the length of the dataset and improving the efficiency of the model. When applied to the Support Vector Machines (SVM), Pearson Correlation and SelectKBest methods resulted the same in train accuracy of 95.60% and in test accuracy of 95.28% respectively. Even comparison using Cross-Validation through K-fold analysis showcased the identical score for both feature selection method; which is about 95.33%. Hence, based on these assessments, both the Pearson Correlation and SelectKBest methods showed identically consistent and marginally superior performance in optimizing the model's predictive capabilities.

4.3 Performance of models with different hyperparameter optimization (on the best model)

In Table 7 shows the SVM model performance remained consistently high, ranging between 95.07% and 95.34% in test accuracy across diverse hyperparameter optimization techniques such as manual ,GridSearchCV, Randomized SearchCV, Sequential Model Based Optimization,Optuna, Genetic Algorithms (TPOT Classifier) and Bayesian Optimization. Meanwhile methods like manual and TPOT Classifier maintained a bit low accuracy than the rest of the methods which are 0.952822 and 0.950732 respectively. As a conclusion, from table 7 we

Type of Hyperparameter Tuning	SVM Train Accuracy	SVM Test Accuracy
Manual	0.956099	0.952822
GridSearchCV	0.955075	0.953419
Randomized SearchCV	0.955075	0.953419
Sequential Model-Based Optimization	0.955075	0.953419
Optuna- Automate	0.955075	0.953419
Genetic Algorithms (TPOT Classifier)	-	0.950732
Bayesian Optimization - Automate Hyperparameter Tuning (Hyperopt)	0.955075	0.953419

Table 7: Hyperparameter Tuning on SVM

can see the other methods showed a slightly improved accuracy to 95.34%.here is the comparison table:

Model	Train Accuracy	Test Accuracy	Train Error Rate	Test Error Rate
SVM (Before Tuning)	0.956099	0.952822	0.043901	0.047178
SVM (After Tuning)	0.955075	0.953419	0.0449245	0.046581

Table 8: comparison table

In Table 8 shows the SVM model, before hyperparameter tuning, achieved a training accuracy of 0.956099 and a test accuracy of 0.952822. After hyperparameter tuning (using methods except manual or Genetic Algorithms (TPOT Classifier)), the training accuracy slightly decreased to 0.955075, but the test accuracy slightly increased to 0.953419. This indicates that the model generalizes better to unseen data after hyperparameter tuning. In terms of error rate, the SVM model had a lower error rate before hyperparameter tuning compared to after. This suggests that while hyperparameter tuning improved test accuracy, it also slightly increased the error rate.

Model	Train Accuracy	Test Accuracy	Train Error Rate	Test Error Rate
Bagging	0.955075	0.952225	0.044925	0.047775
Out of Bag Evaluation	0.955075	0.952225	0.044925	0.047775
Pasting (Best Performing)	0.955075	0.953419	0.044925	0.046581

Table 9: Ensemble Learning – Voting Classifier (Only on SVM)

4.4 Performance of models with Ensemble Learning- Voting Classifiers (on the best model)

Upon analysis from Table 9, this table outlines the comparison of the SVM model's performance metrics after the application of various Ensemble Learning techniques specifically different Voting Classifiers;such as : Bagging, Out of Bag Evaluation(OOB), and Pasting. Notably, the Pasting voting classifier method stood out with a test accuracy of 95.34%, similar to the accuracy obtained from combining all hyperparameter-tuned models. This consistent accuracy, especially with the Pasting method, emphasizes the stability and reliability of ensemble learning in refining the SVM model's predictive capacity.

5 Discussion

This study aims to build a model which presents a promising development in leveraging machine learning techniques for the management of traffic and navigation for autonomous vehicles. Utilizing the Swarm Behaviour Classification dataset, this model incorporates various machine learning methodologies, hyperparameter tuning approaches, and ensemble learning strategies to optimize its performance.

The project entails a series of pivotal steps, from data preprocessing and feature selection to model development, hyperparameter tuning, and ensemble learning techniques. Through a comprehensive evaluation utilizing diverse metrics and methodologies, the model's potential for innovation and advancement within the field becomes evident.

Comparatively, this model distinguishes itself from existing works through its adaptable nature. Unlike many predecessors limited to specific vehicle types, this model demonstrates flexibility for diverse vehicle applications. This broad adaptability stands as a considerable advantage in the domain.

Here is a comparison table that includes our work and other notable works in the field:

Work	Major Limitation	Impact on Traffic Management	How our Model Addresses It
[22]	Data collection	Due to the limited amount of unstructured data,the model is not capable of capturing the complexity of traffic patterns	The dataset we used to train our model has way more comprehensive information in terms of traffic management.
[23]	Real-time adaptability	Depending on the current traffic scenario, the model adjusts the timing of the traffic signal, which might not be able to predict future traffic conditions	Usage of diverse data and the implementation of various machine learning algorithms ensure the potential to anticipate future traffic conditions.
[24]	Dynamic Environment Handling	Usage of deep reinforcement learning to handle dynamic environments might cause instability and reduce its reliability	Using a data-driven approach makes our model better equipped to handle such situations.
[25]	Scalability	No description of how well the model scales, which results in poor performance to manage traffic with large numbers effectively	The machine learning algorithms our model uses make it more reliable for managing a large number of autonomous vehicles
[26]	Simulation-based approach	Lack of capability to capture the complexity under various traffic conditions.	The specific dataset our model uses helps to provide a more accurate representation of real-world traffic scenarios.

From the table, it is clear that while each work has made significant contributions in their respective areas, they are limited in their scope of application or do not consider certain aspects that are addressed by our model. Our model leverages the principles of swarm behaviour and applies them to a broader range of vehicles for both traffic management and navigation. This makes our work stand out among the others.

6 Limitation of the work

This article presents a novel approach to traffic management and navigation for autonomous vehicles using machine learning models. The model was developed using the Swarm Behaviour Classification dataset from UCI. However, the work has experienced a number of challenges:

- The novel approach involved the use of a voting algorithm that was applied on several machine learning models. Consequently, this needed a much more extensive feature set compared to the previous work that exists in the literature.
- Due to the extensive feature set, the model required a large amount of data for training, which was time-consuming and computationally intensive.
- While the model shows good performance in a controlled experimental setup, its ability to perform in real-time traffic management and navigation scenarios needs further testing and validation. This is a common challenge in applying machine learning models to real-world applications.

These limitations present opportunities for further research and improvements in future work. Despite these challenges, the model shows promise in managing traffic and navigation for autonomous vehicles and contributes to the ongoing research in this field.

7 Conclusion and Future Scope

The project successfully developed a machine learning-based system for traffic control and navigation in autonomous vehicles. Utilizing the Swarm Behaviour Classification dataset, diverse machine learning techniques, hyperparameter tuning, and ensemble learning strategies were applied to optimize model performance.

Key findings:

- Machine learning techniques have the potential to significantly improve traffic management efficiency.
- Different tuning and ensemble methods resulted in comparable accuracies with minor variations.

While promising, the work faces limitations in dataset scope, model generalizability, potential overfitting, computational demands, and real-time applicability.

Future directions include:

- Expanding datasets to cover varied real-world traffic scenarios.
- Assessing the model's adaptability across different scenarios and mitigating overfitting concerns.

- Optimizing the model for resource efficiency and testing it in real-time traffic scenarios.

In conclusion, this project offers a novel approach to autonomous vehicle traffic management and presents potential for further development. It marks a substantial stride toward safer and more efficient traffic control and navigation systems for autonomous vehicles.

References

[1] https://www.weforum.org/agenda/2021/01/covid-19-10-most-congested-world-cities-congestion-traffic/These 10 most congested cities in the world — World Economic Forum (weforum.org)

[2] https://link.springer.com/article/10.1007/s42486-020-00039-xDeep learning for intelligent traffic sensing and prediction: recent advances and future challenges — SpringerLink

[3] https://ieeexplore.ieee.org/document/9367130Traffic Flow Management of Autonomous Vehicles Using Deep Reinforcement Learning and Smart Rerouting — IEEE Journals & Magazine — IEEE Xplore

[4] https://www.brookings.edu/articles/traffic-why-its-getting-worse-what-government-can-do/Traffic: Why It's Getting Worse, What Government Can Do — Brookings

[5] http://encyclopedia.uia.org/en/problem/urban-road-traffic-congestionUrban road traffic congestion — The Encyclopedia of World Problems (uia.org)

[6] https://archive.ics.uci.edu/dataset/524/swarm+behaviourUCI Machine Learning Repository

[7] https://www.kaggle.com/datasets/deepcontractor/swarm-behaviour-classificationSwarm Behaviour Classification (kaggle.com)

[8] https://www.analyticsvidhya.com/blog/2018/06/comprehensive-guide-for-ensemble-models/Ensemble Learning — Ensemble Techniques (analyticsvidhya.com)

[9] https://link.springer.com/article/10.1007/s41062-021-00718-3A comprehensive review on intelligent traffic management using machine learning algorithms — SpringerLink

[10] https://link.springer.com/chapter/10.1007/978-3-662-48847-$8_1$6

[11] https://www.recogni.com/driverless-cars-and-traffic-efficiency/

[12] https://ieeexplore.ieee.org/document/9231360/

[13] https://ce.berkeley.edu/news/2537

[14] https://www.thalesgroup.com/en/markets/digital-identity-and-security/iot/magazine/7-benefits-autonomous-cars

[15] https://link.springer.com/article/10.1007/s41062-021-00718-3

[16] https://ieeexplore.ieee.org/document/9482842

[17] https://link.springer.com/chapter/10.1007/978-3-030-97546-3$_4$3

[18] https://link.springer.com/chapter/10.1007/978-3-030-74540-0$_5$

[19] https://muthu.co/understanding-the-classification-report-in-sklearn/: :text=A

[20] https://hasty.ai/docs/mp-wiki/metrics/accuracy: :text=The

[21] https://www.sciencedirect.com/topics/engineering/confusion-matrix: :text=A

[22] https://arxiv.org/abs/2206.14608: :text=28

[23] https://www.researchgate.net/publication/330858371$_{A}daptive_{T}raffic_{M}anagement_{S}ystem_{U}sing_{I}oT_{a}nd$

[24] https://ieeexplore.ieee.org/abstract/document/9868790

[25] https://www.sciencedirect.com/science/article/pii/S2666827021000827

[26] https://www.sciencedirect.com/science/article/abs/pii/S0925231222006427

[27] https://www.sciencedirect.com/science/article/abs/pii/S0263224120304231

[28] https://einj.net/index.php/INJ/article/view/173