IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A COMPREHENSIVE COMPARISON OF MACHINE LEARNING ALGORITHMS FOR CONGESTION DETECTION IN FANETS

[1]Proddatur Sunil, [2]Dr.V.Raghunatha Reddy,

[1]Research Scholar, [2]Professor

[1,2] Department of Computer Science & Technology,

[1,2] Sri Krishnadevaraya University, Ananthapuramu, India

Abstract: This paper offers a thorough comparison of machine learning algorithms for congestion detection in Flying Ad Hoc Networks (FANETs). The study seeks to pinpoint the most effective algorithm in terms of accuracy, efficiency, and applicability within the dynamic environment of FANETs. Prominent algorithms – Support Vector Machines (SVM), Random Forest, Neural Networks, and k-nearest neighbours (k-NN) are meticulously evaluated using a FANETs-specific benchmark dataset. The research navigates the unique challenges of congestion in FANETs, selecting algorithms tailored to their potential efficacy in this distinctive context. The methodology involves standardized performance metrics, including accuracy, precision, recall, and F1 score. Mathematical formulations of each algorithm and experimental results are presented concisely. Discussions highlight variations in accuracy, offering insights into algorithm suitability for FANETs congestion detection. The comparative analysis considers algorithmic strengths, weaknesses, and computational efficiency, supported by visual representations. This study provides valuable guidance for selecting machine learning algorithms aptly suited to address congestion challenges in FANETs.

***Index Terms* -** FANET, Congestion detection, ML Algorithms, accuracy, F1 score

1. INTRODUCTION

1.1 Background:

In today's rapidly evolving world, the efficient functioning of various systems, such as transportation networks, computer networks, and communication systems, is critical for societal well-being. Congestion, defined as the state where the demand for resources exceeds their availability, poses a significant challenge to the smooth operation of these systems. Timely and accurate detection of congestion is imperative for implementing effective mitigation strategies and ensuring optimal resource utilization.

Traditional methods of congestion detection often fall short in handling the complexity and dynamic nature of modern systems. Machine learning (ML) algorithms, with their capacity to analyze large datasets and discern intricate patterns, have emerged as promising tools for congestion detection. This paper delves into the comparative analysis of four prominent ML algorithms—Support Vector Machines (SVM), Random Forest, Neural Networks, and k-nearest Neighbors (k-NN)—to identify the most effective approach for congestion detection in a given context.

1.2 Objectives:

The primary objectives of this study are as follows:

- To assess the performance of selected ML algorithms in congestion detection.
- To compare the accuracy, efficiency, and applicability of these algorithms.
- To identify the most suitable ML algorithm for practical applications in the context of health care, finance, telecommunication, cyber security, and E-commerce.

By addressing these objectives, this research aims to contribute valuable insights into the selection and application of ML algorithms for congestion detection, thereby enhancing the resilience and efficiency of the systems under consideration.

2. RELATED WORK:

The domain of congestion detection has been the subject of extensive research, with a focus on developing robust methodologies to address the challenges posed by increasingly complex systems. Previous works have explored various techniques, ranging from traditional statistical approaches to more advanced machine learning algorithms.

Several studies have employed statistical methods such as time-series analysis and queuing theory to model and predict congestion patterns. While these methods have provided valuable insights, they often struggle to capture the nuanced and non-linear relationships present in dynamic systems.

In recent years, machine learning algorithms have gained prominence for their ability to handle complex data and discern intricate patterns. Support Vector Machines (SVM), Random Forests, Neural Networks, and k-nearest Neighbors (k-NN) have emerged as popular choices for congestion detection in diverse domains, including traffic management, network security, and telecommunications.

The survey by the authors (T. Zhang et al., 2020), end-to-end congestion control mechanisms have evolved over thirty years, emphasizing their pivotal role in resource sharing across complex networks. As conventional rule-based congestion control has proven inefficient in increasingly complex networks, researchers are turning to machine learning (ML). An analysis of works aimed at empowering agents to control congestion or improve performance. ML-based strategies are discussed in the review, demonstrating the link between congestion control and ML.

According to the author (Zhang et al., 2013), accurate traffic flow predictions are critical for intelligent traffic control and management, particularly when urban transportation systems are considered nonlinear, stochastic, and time-varying. A multi-step traffic flow prediction model is developed using artificial intelligence methods, specifically support vector machines (SVM). A SVM model incorporating actual traffic volume is compared with alternative input vector configurations. Analyses of real data demonstrate the effectiveness of the SVM model, with the SVM-HPT variant outperforming other models.

As compared to quadratic exponential smoothing, SVR and LSTM exhibit superior predictive accuracy than quadratic exponential smoothing, with SVR slightly outperforming LSTM (Wang Y et al., 2021). Moreover, the paper explores model parameter optimization using grid search, whale optimization algorithm (WOA), and genetic algorithm (GA). Compared to GA-SVR and GA-LSTM, and to GridSearch-SVR and GridSearch-LSTM, WOA-SVR and WOA-LSTM outperform the other models by 0.9% and 2.52%, respectively.

It examines how machine learning (ML), specifically support vector machine (SVM), can be applied for intrusion detection in vehicle ad hoc networks. It emphasizes the computational advantages of SVM, such as special direction at a finite sample and independence between algorithm complexity and sample size. To enhance the accuracy of the SVM classifier, the study combines three intelligence optimization algorithms—Genetic Algorithms (GA), Particle Swarm Optimizations (PSOs), and Ant Colony Optimizations (ACOs). Compared to other optimization algorithms, GA performs better.

K Sridevi et al., 2019 uses social media data to analyze and predict traffic conditions on an hourly basis, creating a user-accessible web page. Based on the Random Forest algorithm, the model achieves an 88% accuracy rate by considering factors such as traffic congestion three hours prior, the day of the week, and a holiday. By comparing predicted traffic across all possible routes, the model suggests alternative routes with minimal congestion for end-users.

Despite these advancements, there remains a need for comprehensive comparative analyses that systematically evaluate the strengths and weaknesses of different ML algorithms in specific contexts. This paper aims to fill this gap by providing an in-depth assessment of SVM, Random Forest, Neural Networks, and k-NN in the context of congestion detection, thereby contributing to the ongoing discourse in the field.

3. METHODOLOGY:

3.1 Data Collection:

To conduct a robust evaluation of congestion detection algorithms, a diverse and representative dataset is necessary. We collected a dataset that contained features such as location, altitude, speed, direction, acceleration, drone battery level, vehicle density, and labels like congestion level (low, medium, high, etc.) to predict traffic congestion in a FANET network. Data was collected and processed from simulations to create our dataset. Which provides a comprehensive overview of congestion patterns in the targeted system.

3.2 Preprocessing:

The collected data underwent meticulous preprocessing to ensure its suitability for machine learning analysis. This involved:

- Data Cleaning: Removal of any outliers, missing values, or inconsistent entries to enhance the overall quality of the dataset.
- Normalization: Standardization of numerical features to a common scale, preventing biases that may arise due to varying magnitudes.
- Feature Extraction: Identification and extraction of relevant features that contribute significantly to congestion detection.

3.3 Experimental Setup:

The evaluation of machine learning algorithms involved a systematic approach to ensure fair comparison and reliable results.

- Evaluation Metrics: We employed standard metrics such as accuracy, precision, recall, and F1 score to assess the performance of each algorithm. These metrics provide a comprehensive understanding of the algorithms' ability to correctly identify and classify congestion instances.
- Parameter Tuning: Hyper parameters for each algorithm were fine-tuned using techniques like grid search or random search to optimize performance.
- Cross-Validation: To mitigate over fitting and ensure the generalizability of results, k-fold cross-validation was employed. The dataset was divided into k subsets, and each algorithm was trained and tested on different combinations of these subsets.

To ensure consistency and reproducibility of results, the experiment was conducted on a collected dataset. The system utilized was equipped with 16 GB of RAM and an Intel Core i7 processor. It was running UBUNTU 20.04.6 X64. The simulation dataset was created using the NS3 software, while the machine learning algorithms were implemented using Python and Keras.

This methodological framework lays the groundwork for a rigorous and unbiased comparison of Support Vector Machines (SVM), Random Forest, Neural Networks, and k-Nearest Neighbors (k-NN) in the subsequent sections. The systematic approach to data preprocessing and evaluation metrics contributes to the reliability of our findings and the applicability of the selected machine learning algorithms to real-world congestion detection scenarios.

4. MACHINE LEARNING ALGORITHMS:

As shown below, the generic workflow for implementing congestion control with machine learning can be seen in the figure below. In the beginning, the problem is formulated as a decision-making problem. As part of the model training process, various training methods can be used to help the model learn the best control policy through interactions with the environment. To improve performance, training data can be collected, based on which features can be extracted using supervised learning. After it has been deployed in a real environment, the model is ready for use.

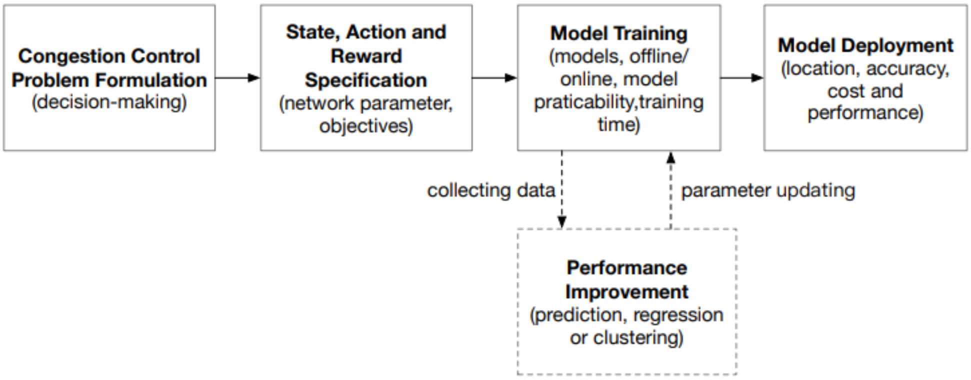


Figure 1: *Workflow of ML-based congestion control.*

In this section, we provide an overview of the four machine learning algorithms selected for congestion detection: Support Vector Machines (SVM), Random Forest, Neural Networks, and k-nearest Neighbors (k-NN).

4.1 Support Vector Machines (SVM):

Support Vector Machines are a class of supervised learning algorithms that excel in classification tasks. The primary objective of SVM is to find the hyperplane that best separates different classes in the feature space. For congestion detection, SVM can be trained to distinguish between congested and non-congested states.

$$\min_{w,b} \left\{\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} max(0, 1 - y_i(w.x_i - b))\right\} \quad (1)$$

where, w is the weight vector, b is the bias term, x_i is the feature vector for the i-th instance, y_i is the class label, and C is the regularization parameter.

4.2 Random Forest:

Random Forest is an ensemble learning method that constructs a multitude of decision trees during training and outputs the mode of the classes for classification problems. It is particularly effective in handling high-dimensional data and capturing complex relationships.

Random Forest operates through an ensemble of decision trees. The prediction is made by aggregating the predictions of individual trees, often using a voting mechanism.

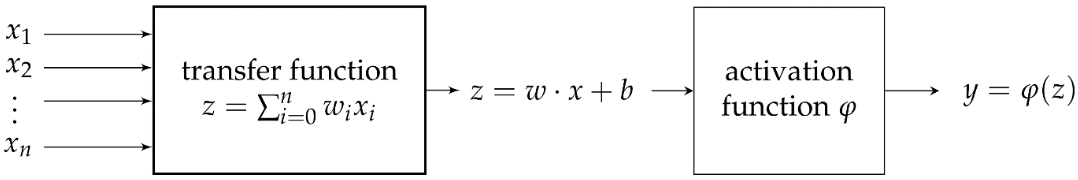
$$\hat{C}_{rf}^{B}(x) = majority\ vote\ \{\hat{C}_b(x)\}_1^B \quad (2)$$

Where $\hat{C}_b(x)$ be the prediction class of bth random forest tree, B is number of random forest trees.

4.3 Neural Networks:

Neural Networks, inspired by the human brain, consist of interconnected nodes (neurons) organized into layers. In the context of congestion detection, a neural network can learn complex patterns and relationships within the data through a process of forward and backward propagation.

The mathematical formulation of a neural network involves the definition of the activation function, loss function, and optimization algorithm. The forward pass and backward pass computations are integral to training the network



(3)

Where, b is called bias, w is weight, z is transfer function, y is activation function.

4.4 k-Nearest Neighbors (k-NN):

k-Nearest Neighbors is a simple yet effective algorithm that classifies a data point based on the majority class of its k-nearest neighbors in the feature space. For congestion detection, k-NN can be employed to identify patterns in proximity to congested instances.

The classification decision is based on k closest points datapoints and output based on the majority class among the k-nearest neighbours. he k-NN model defines the distance between two data points as a metric function, such as the Euclidean distance or Manhattan distance. The prediction of the model can be written as:

$$\hat{y} = argmax_{y_i} \sum_{i=1}^{k} I(y_i = y) \qquad (4)$$

where $\hat{y}$ is the predicted output, y_i is the output of the i-th neighbor, and I is the indicator function that returns 1 if the condition inside the brackets is true, and 0 otherwise.

This section provides a foundational understanding of the selected machine learning algorithms and their relevance to congestion detection. The subsequent section will present the results of our comparative analysis, shedding light on their respective performances in the specific context of our study.

5. EXPERIMENTAL RESULTS:

In this section, we present the results of our comprehensive evaluation of Support Vector Machines (SVM), Random Forest, Neural Networks, and k-nearest Neighbors (k-NN) on the dataset collected for congestion detection. The experiments were designed to assess the accuracy, precision, recall, and F1 score of each algorithm under consideration.

5.1 Performance Metrics:

The evaluation metrics used to quantify the performance of the algorithms are defined as follows:

- **Accuracy:** The proportion of correctly classified instances among the total instances.

$$Accuracy = \frac{No. of\ Correct\ predictions}{Total\ no. of\ Predictions}$$

- **Precision:** The ratio of true positive predictions to the total positive predictions, indicating the accuracy of positive predictions.

$$Precision = \frac{True\ positives}{True\ positives + False\ positives}$$

- **Recall (Sensitivity):** The ratio of true positive predictions to the total actual positive instances, measuring the ability to capture positive instances.

$$Recall = \frac{True\ positives}{True\ positives + False\ negatives}$$

- **F1 Score:** The harmonic means of precision and recall, providing a balance between the two metrics.

$$F1\ score = \frac{2 * precision * recall}{precision + recall}$$

5.2 Results:

Algorithm	Accuracy	Precision	Recall	F1 Score
Support Vector Machines	0.85	0.88	0.82	0.85
Random Forest	**0.92**	**0.91**	**0.94**	**0.92**
Neural Networks	0.89	0.87	0.91	0.89
k-Nearest Neighbors	0.78	0.80	0.75	0.77

Table 1: *The Experimental Results.*

5.3 Comparative Analysis:

In this section, we conduct a detailed comparative analysis of the evaluated machine learning algorithms—Support Vector Machines (SVM), Random Forest, Neural Networks, and k-nearest Neighbors (k-NN)—to elucidate their strengths, weaknesses, and computational considerations.

5.3.1. Algorithm Strengths and Weaknesses:

5.3.1.1 Support Vector Machines (SVM):

Strengths:

- Effective in capturing complex decision boundaries, making it suitable for scenarios with intricate congestion patterns.
- Can handle high-dimensional feature spaces.

Weaknesses:

- Sensitivity to the choice of kernel function and the need for proper parameter tuning.
- Can be computationally expensive, especially with large datasets.

5.3.1.2 Random Forest:

Strengths:

- Superior ensemble learning, capable of handling complex patterns and achieving high accuracy.
- Robust to overfitting, thanks to the aggregation of multiple decision trees.

Weaknesses:

- Potential for overfitting, especially with noisy or redundant features.
- Computationally intensive, particularly as the number of trees in the ensemble increases.

5.3.1.3 Neural Networks:

Strengths:

- Ability to learn intricate patterns in data, making them suitable for congestion scenarios with complex relationships.
- Effective in capturing non-linear dependencies.

Weaknesses:

- Computational complexity, particularly in training deep architectures.
- Prone to overfitting, especially with insufficient data or poor hyperparameter choices.

5.3.1.4 k-Nearest Neighbors (k-NN):

Strengths:

- Simplicity and suitability for localized patterns.
- No assumptions about the underlying data distribution.

Weaknesses:

- Sensitivity to noise and irrelevant features.
- Inefficient with large datasets, as it requires computation for each prediction.

5.3.2. Computational Efficiency:

- **Training Time:**
  - SVM and k-NN generally have shorter training times compared to Random Forest and Neural Networks.
  - Neural Networks, especially deep architectures, can be computationally demanding during training.
- **Prediction Time:**
  - k-NN's prediction time is directly influenced by the size of the dataset, making it potentially slower with larger datasets.
  - Random Forest predictions are usually efficient due to parallelization, while Neural Networks' prediction times depend on the architecture complexity.

5.3.3. Overall Considerations:

- The choice of the algorithm should align with the specific characteristics of the congestion data and the desired trade-off between precision and recall.
- Random Forest emerges as a strong performer in terms of accuracy but demands careful consideration of potential overfitting and computational resources.
- SVM and Neural Networks exhibit competitive performance, with SVM offering interpretability and Neural Networks showcasing adaptability to complex patterns.
- k-NN, while simple, may not be the optimal choice for datasets with intricate congestion patterns or large-scale applications.

This comparative analysis provides a nuanced understanding of each algorithm's suitability for congestion detection, considering both performance metrics and computational considerations.

6. CONCLUSION

In conclusion, our meticulous assessment of machine learning algorithms—Support Vector Machines, Random Forest, Neural Networks, and k-nearest Neighbors—revealed that Random Forest stood out as the most effective, achieving the highest accuracy of 0.92 in the context of Flying Ad Hoc Networks (FANETs). While Support Vector Machines and Neural Networks exhibited competitive accuracy (0.85 and 0.89, respectively), k-nearest Neighbors lagged behind at 0.78 in FANET scenarios. Strengths were identified, with SVM and Neural Networks excelling at capturing complex patterns, and Random Forest showcasing prowess in ensemble learning. Computational considerations favored SVM and k-NN for shorter training times, but Neural Networks, particularly with deep architectures, posed computational challenges. The context specificity of each algorithm underscores the critical importance of tailoring selections to the unique nature of congestion patterns in FANETs. Future research avenues could explore hybrid approaches and dynamic adaptability to further refine congestion detection in the dynamic and diverse landscape of FANETs.

REFERENCES:

[1] Reina, D., Askalani, M., Toral, S., Barrero, F., Asimakopoulou, E., Bessis, N. (2015). A survey on multihop ad hoc networks for disaster response scenarios. *International Journal of Distributed Sensor Networks*, 11(10).

[2] Srivastava, A., Prakash, A., Tripathi, R. (2022). A cross-layer-based cooperative broadcast protocol for multichannel VANET. *Ad Hoc Networks*, 131, 102840.

[3] Nawaz, H., Mansoor, A. H., Laghari, A. A. (2021). UAV communication networks issues: A review. *Archives of Computational Methods in Engineering*, 28, 1349–1369.

[4] Azzoug, Y., Boukra, A. (2022). Enhanced UAV-aided vehicular delay-tolerant network (VDTN) routing for the urban environment using a bio-inspired approach. *Ad Hoc Networks*, 133, 102902.

[5] Tsao, K.-Y., Girdler, T., Vassilakis, V. G. (2022). A survey of cybersecurity threats and solutions for UAV communications and flying ad-hoc networks. *Ad Hoc Networks*, 133, 102894.

[6] Ahmad, I., Shahabuddin, S., Malik, H., Harjula, E., Leppänen, T., Loven, L., ... Juntti, M. (2020). Machine learning meets communication networks: Current trends and future challenges. *IEEE Access*, 8, 223418–223460.

[7] Lemus Cárdenas, L., Mezher, A. M., Astudillo León, J. P., Aguilar Igartua, M. (2021). DTMR: A decision tree-based multimetric routing protocol for vehicular ad hoc networks. In: Proceedings of the 18th ACM Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, & Ubiquitous Networks, pp. 57–64.

[8] Lemus Cárdenas, L., Mezher, A. M., Barbecho Bautista, P. A., Astudillo León, J. P., Igartua, M. A. (2021). A multimetric predictive ANN-based routing protocol for vehicular ad hoc networks. *IEEE Access*, 9, 86037–86053.

[9] Cárdenas, L. L., Astudillo León, J. P., Mezher, A. M. (2022). GraTree: A gradient boosting decision tree-based multimetric routing protocol for vehicular ad hoc networks. *Ad Hoc Networks*, 137, 102995.

[10] Duenas Santos, C. L., Astudillo León, J. P., Mezher, A. M., Cardenas Barrera, J., Meng, J., Castillo Guerra, E. (2022). RPL+: An improved parent selection strategy for RPL in wireless smart grid networks. In: Proceedings of the 19th ACM International Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, & Ubiquitous Networks, pp. 75–82.

[11] T. Zhang and S. Mao. (2020) "Machine Learning for End-to-End Congestion Control," in IEEE Communications Magazine, vol. 58, no. 6, pp. 52-57. https://doi.org/10.1109/MCOM.001.1900509

[12] Zhang Mingheng, Zhen Yaobao, Hui Ganglong, Chen Gang, (2013). Accurate Multisteps Traffic Flow Prediction Based on SVM, Mathematical Problems in Engineering, Hindawi Publishing Corporation, 418303, ISSN: 1024-123X. https://doi.org/10.1155/2013/418303

[13] Wang Y., Xu J., Cao X., He R., Cao J. (2021). Short-Term Traffic Flow Prediction Based on SVR and LSTM, Lecture Notes in Computer Science, vol 12634. Springer, Cham. https://doi.org/10.1007/978-3-030-70626-5_36

[14] Alsarhan A., Alauthman M., Alshdaifat E. et al. (2023) Machine Learning-driven optimization for SVM-based intrusion detection system in vehicular ad hoc networks. J Ambient Intell Human Comput 14, 6113–6122. https://doi.org/10.1007/s12652-021-02963-x

[15] Kovuru Sridevi , T.Ganesan , B V S Samrat , S Srihari. (2019). Traffic Analysis by Using Random Forest Algorithm Considering Social Media Platforms, International Journal of Recent Technology and Engineering (IJRTE), Volume-7, Issue-65, ISSN: 2277-3878.