Proceedings of the International Conference on Industrial Engineering and Operations Management
Publisher: IEOM Society International, USA DOI: 10.46254/NA10.20250208
Published: June 17, 2025

AI-Based Solutions for Optimizing Lane Change Safety in Connected Automated Vehicles

Chase Waelbroeck and Rawa Adla
Department of Electrical and Computer Engineering
Florida Polytechnic University
Lakeland, FL, USA
cwaelbroeck@floridapoly.edu, radla@floridapoly.edu

Abstract

Connected Automated Vehicles (CAVs) have the potential to enhance safety and mobility by reducing the number of crashes on the road and improving the real-time traffic flow. CAV technologies enable vehicles to communicate and exchange information with other vehicles, pedestrians and infrastructure utilizing the Vehicle-to-everything (V2X) communication. Different data processing methodologies were implemented using these data enhance the decision making. However, with the advancements of Artificial Intelligence (AI), research results indicate that AI plays a crucial role in optimizing decision-making within the CAV networks. This study analyzes and categorizes the AI techniques such as Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) in CAV environments. It also provides insights into the effectiveness of each approach in different traffic applications. This research contributes to the ongoing development of AI-enhanced CAV system for safer and efficient intelligent transportation systems.

Keywords
Intelligent transportation system, Connected vehicles, artificial intelligence, neural networks, and safety.

1. Introduction
According to the World Health Organization 2023 report on global road safety, there were an estimated 1.19 million road traffic deaths in 2021, which corresponds to a rate of 15 road traffic deaths per 100,000 population. Traffic continues to increase, and more vehicles are on the road than ever. Transportation requires continual progress on innovative technologies that allow for safer roadways and a more efficient traffic flow. CAVs improve roadway safety and traffic flow. CAVs combine two separate processes: connected vehicles and autonomous vehicles. Rana and Hossain (2021) wrote that progress is focused on the future of Connected Vehicles using communication technologies to establish connections with autonomous and non-autonomous vehicles, roadside infrastructures, and other road participants to share driving information. Autonomous Vehicles, also known as driverless vehicles, can move from one place to another using sensors and a communication module without any human intervention. Both can work in tandem, allowing CAVs to implement both functions.

As research continues to develop and improve upon CAVS, a strong focus has been placed on machine learning and neural networks. Micheli et al. (2013) paper covered that machine learning and artificial intelligence (AI) is a popular approach in the present research environment as it deals with various applications in image processing, signal processing, and other data analysis. In a CAV environment, the need for data to be processed accurately and efficiently is paramount due to the expense that these calculations produce. Yang et al. (2021) state that it is prohibitively expensive to transmit raw data among vehicles, causing network congestion, packet drops, and large processing delays. Paths and feature maps generated by the machine learning models on autonomous vehicles can be viewed as an alternative representation of the original sensor data and thus can be transmitted and fused to effectively realize cooperative perception on CAVs. Chen (2019) proves that paths and feature maps generated on two vehicles can be combined to realize more accurate cooperative object detection. These paths and feature maps are generated by Artificial Neural Networks (ANNs), a more advanced form of AI with specialized architectures.

Recurrent Neural Networks (RNN) and Convolutional Neural Networks (CNN) have been recognized as important architectures for CAVs. RNNs excel at processing sequential data where the order of elements is important for its understanding. While CNNs can excel at capturing hierarchical patterns and spatial dependencies. Both neural networks have strengths that can be leveraged for different applications, these strengths can be further enhanced with the addition of Long short-term memory (LSTM). As Lindemann et al. (2021) states LSTM gating system contributes to the filtering of irrelevant input information and achieves a higher precision in the modeling of time-variant behavior. LTSM has been used for dynamic system modeling such as autonomous systems for great benefits.

1.1 Objectives

This research study aims to evaluate and analyze the traffic throughput and safety applications in CAVs utilizing machine leaning techniques and propose the optimal solution for traffic flow and safety. The focus is on convolutional neural networks (CNNs) and recurrent neural networks (RNNs) each utilizing a long short-term memory (LSTM) and how they can assist in deciding safe lane changes. Unlike prior studies that focus solely on temporal or spatial features in lane change optimization, this study provides a direct comparative analysis of RNN-LTSM and CNN-LSTM models for lane change optimization, focusing on their strengths across different input states. In Section 2, an overview of AI used in a CAV environment will be reviewed and discussed in this paper. Section 3 defines the methods that will be applied in this paper. Section 4 goes into detail on the data utilized and how it was managed. Section 5 discusses the results and analysis of the model. Section 5.1 goes into proposed improvements to be conducted. Section 5.2 explains how the model and data was validated. Section 6 will include the closing remarks of this paper, and the objectives covered throughout.

2. Literature Review

CAVs are vehicles that make driving choices based on sensors and network connectivity to interact with the environment without human interaction. As stated by He et al. (2019) CV uses vehicles to everything (V2X) communication technology to communicate with other road users and networks, including V2V, V2P and V2I. With Vehicle to Vehicle (V2V) On-Board Units (OBUs) communicate if they are in proximity to each other exchanging relevant data for processing. Vehicle to Pedestrian (V2P) information is transmitted by the OBU and equipment utilized by pedestrians, with warnings being communicated either way. Vehicle to Infrastructure (V2I) utilizes the OBU to communicate with a Road Side Unit (RSU) or other relevant infrastructure. In addition to Vehicle to Network (V2N) where the OBU communicates to a remote server.

Described by Molina-Masegosa (2020) optimizing traffic flow and collision mitigation are some of the most important target application areas that V2X communications, and accordingly connected vehicles, aim to support. V2X communication enables CAVs to exchange information with their environment, while this ability to interpret and act on this data in real time is driven by Artificial Intelligence (AI). By integrating AI with V2X communication, autonomous vehicles can improve traffic flow and enhance safety. AI is capable of processing data gathered by the sensors on the vehicle and the OBU which can then be used to make driving decisions without human intervention. However, traditional AI approaches can struggle with the complexity that real-world driving environments provide. Artificial Neural Networks (ANNs) address this issue by advancing AI and allowing an enhancement in decision making.

ANNs are similar to the function of a human nervous system. The ANN has processing nodes that are similar to neurons. To simplify, they are simple processors that link together with weighted connections. Dongare et al. (2012) states that each node output depends only on the information that is locally available at the node, either stored internally or arriving via the weighted connections. These node connections allow inputs to feed through the network into outputs continually passed along the model. ANN uses this to process information and update the weights to get the best solution to the wanted approximation. This method trains the neural network to get the desired output based on the input and parameters set. To professionally train the output, the ANN must have a set of outputs for which the correct outputs are known. This section will detail the types of neural networks and enhancements that are applied to model traffic applications and make autonomous driving decisions. Fig. 1 shows the types of ANN and how they differentiate from each other. A feed forward network has the signals it sends only moving in one direction. While a feedback network features feed-back paths, which allows the system to travel in both directions of the neural connections. As

Ford (1999) states feedback is defined to occur when the (full or partial) output of a system is routed back into the input as part of an iterative cause-and-effect process. ANNs serves as a foundation for applications in autonomous vehicles. Traditional ANNs struggle with processing and interpreting visual data. To address this, CNNs have been developed as a specialized type of ANN designed for image and video analysis.

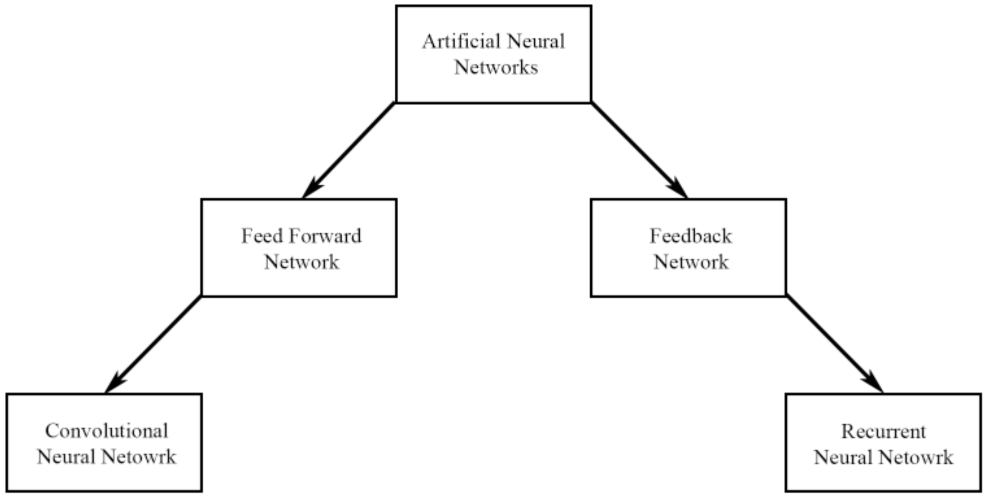

Figure 1. The difference between CNN and RNN networks.

CNN utilizes images and videos to allow for raw data processing throughout the network. Sensors capture images by collecting real-time environmental data. Then compared to internal sensors that measure the vehicles position, orientation, and movement. Younesi et al. (2024) states the CNN architecture typically consists of an initial input layer, followed by several critical components, including convolutional layers, pooling layers, and fully connected layers. CNN architecture functions by utilizing an operation known as convolutions. Younesi et al. (2021) states that convolutions form the foundation of crucial mathematical operations used to process data structured in grids, such as images, videos, and time series data. Taking a small part of an image and sliding the kernel through the entirety of the image is a convolution. Liang et al. (2019) describes that each position of this sliding operation, the kernel performs element-wise multiplication with the corresponding input values. When applying a convolution kernel of a specific size, portions of the border information are lost. Li (2021) describes that padding is introduced to enlarge the input with zero value, which can adjust the size indirectly. The procedure of a CNN is shown in Figure 2.

Building a CNN architecture requires a sequence of layers, and every layer transforms one volume into another through a differentiable function. In a CNN, raw data enters through the input layer and processes in the convolution layer before being passed to the padding layer. Padding is a technique that avoids the loss of information on the input data borders, caused by the kernel operations, and consists of adding zeros around the input margins as described by Rala et al. (2021), which means that all pixels are equally considered throughout the image. The activation layer is what adds nonlinearity to the network. Otherwise, a CNN would not be able to manage more complicated functions. The pooling layer then occurs; in pooling operations, the goal is always to reduce the spatial size of the convolved features and avoid overfitting. The final layer, or the dropout layer, is crucial for reducing the error of the analyzed raw data. Srivastava et al. (2014) found that training a network with dropout and using this approximate averaging method at test time leads to significantly lower generalization errors on a wide variety of classification problems compared to training with other regularization methods. CNNs alone can struggle to understand traffic patterns and sequential data rather than images. RNNs are different than CNNs as they are designed to be a specialized ANN that recognizes patterns in sequential data.

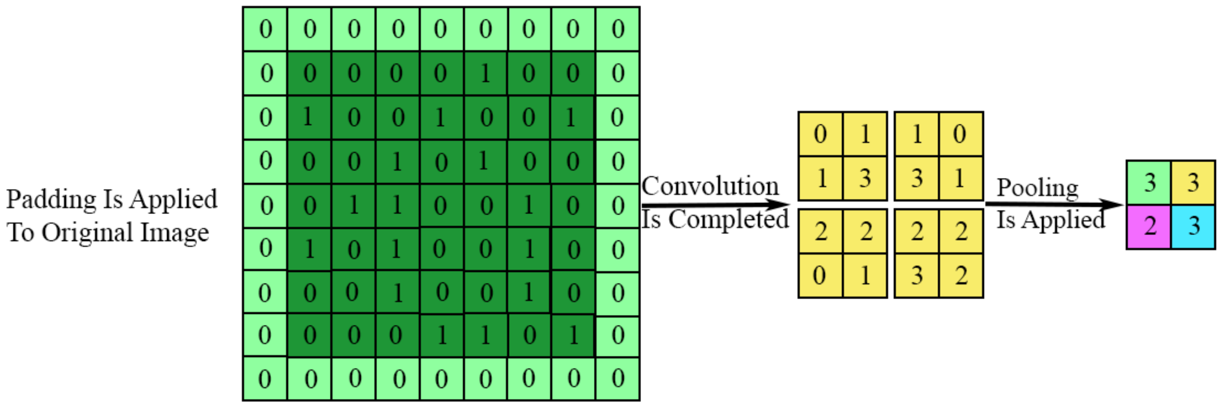

Figure 2. The procedure of a 2-D in CNN.

As described by Das et al. (2023) RNN is a specialized neural network with feedback connection for processing sequential data or time-series data in which the output obtained is fed back into it as input along with the new input at every time step. This network allows the use of past data to influence and advise on how the subsequent output will be. Lipton et al. (2014) stated like feedforward networks, RNNs may not have cycles among conventional edges. However, edges that connect adjacent time steps, called recurrent edges, may form cycles, including cycles of length one that are self-connections from a node to itself across time. The RNN utilizes hidden states that store the sequence information from up to one time step ago, which are inputs to the following given step. The hidden state of an RNN can capture historical information of the sequence up to the current time step. With recurrent computation, the number of RNN model parameters does not grow as the number of time steps increases. Fig. 3 visualizes a simple recurrent neural network RNN. The input layer receives data from a single time step in the sequence. The hidden layer processes the input and has a feedback loop that allows the network to process information from previous time steps. The output layer then generates predictions based on both the input and learned information, passing them forward to the next time step.

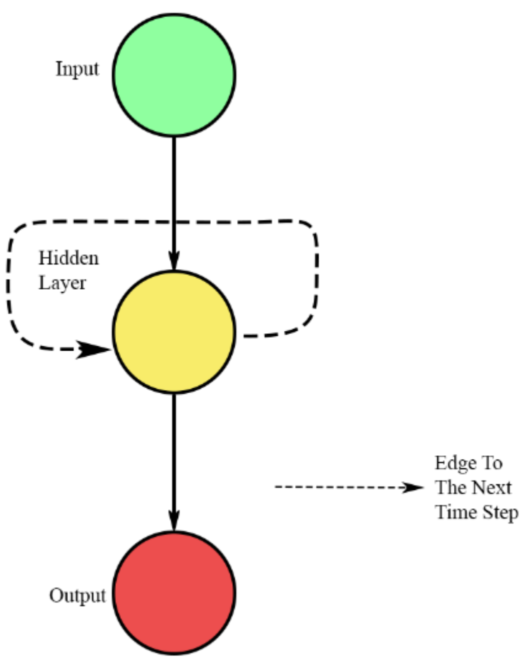

Figure 3. A simple recurrent neural network.

To expand the capabilities of a typical ANN, one effective approach is by introducing a sequential structure dynamically. One method that can accomplish this is by using Long Short-Term Memory (LSTM) architecture, introduced by Hochreiter and Schmidhuber (1995). As noted by Arpit et al. (2018) LSTM has been shown to achieve exceptional success due to its capability of learning both short- and long-term dependencies of the problem and is also designed to deal with the vanishing gradient problem which most of the RNN architectures suffer from. LSTM employs a cell state and gating mechanisms, which provide easier control over the flow of information through the network. As described by Shertinsky (2020) The key principle of the LSTM cell centers around organizing its internal operations according to two qualitatively different, yet cooperating, objectives: data and the control of data. These mechanisms enable LSTMs to effectively capture long term-dependencies, making them highly useful for extracting information from sequential data.

3. Methods

The methodology for this study consists of several steps. The first step is dataset processing after data collection. Data is normalized when necessary to enhance model convergence. This has the normalization step of categorizing the Lane ID to prevent introducing a bias into the models. This allows the model to learn lane-specific patterns without misinterpreting the numerical data. The dataset is then split into a training set (80%) and a test set (20%), which are used to develop the neural networks, as shown in Figure 4.

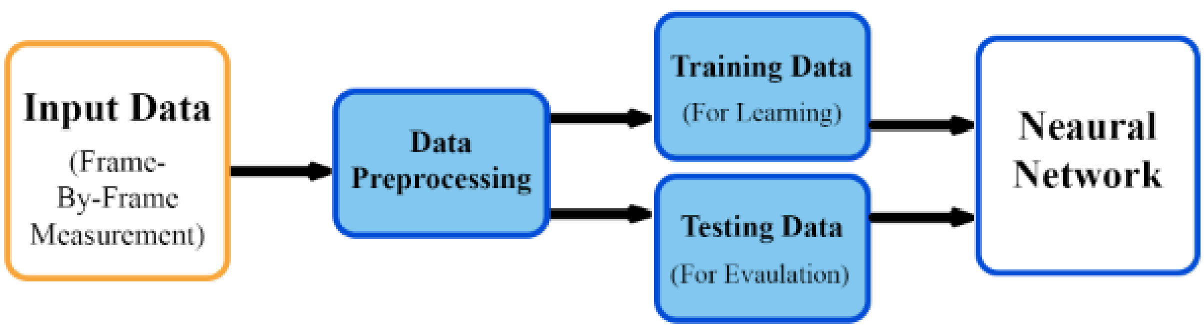

Figure 4. Utilization of data in the neural network

The two models observed are RNN and CNN. Both models utilize a unit called Long Short-Term Memory, otherwise known as LSTM. LSTMs enhance neural networks by providing memory retention and controlled information flow.

This study selected RNN-LSTM and CNN-LSTM architectures due to their complementary strengths in handling sequential and spatiotemporal data, respectively. The RNN-LTSM is well suited for processing temporal dependencies in data communicated from infrastructure and vehicles around it. While CNN-LSTM combines spatial feature extraction with temporal modeling making it appropriate to handle frame by frame views of the roadway surrounding the vehicle. Comparing these architectures allows for an effective assessment of relative accuracy, computational efficiency, and suitability for real time lane change optimization in connected autonomous vehicles. The models have sixteen state dimensions allowing for the size of the vector at each internal time step. Every sequence of the input has fifty frames that the model processes for the context of the vehicle. Each model contains 3 LSTM layers stacked on top of each other with each receiving the output from the previous layer as an input. While both models utilize LSTM models, The CNN-LSTM includes additional convolutional layers for spatial feature extraction, this increases its computational cost compared to the RNN-LSTM model that operates on time-series sequence data. This additional computational cost is expected to raise the training time of the CNN-LSTM model as it trains and tests on the dataset.

To deduce loss the model uses Cross-Entropy Loss was used as the primary objective function for multi-class classification. As stated by Shoham and Permuter (2020) the algorithm for Cross–Entropy Loss is as follows:

$$CrossEntropy\ Loss = -\sum_{i=1}^{C} \gamma_i log\ (\rho_i)$$

Where C is the number of classes for classification, γ_i is the true probability distribution and ρ_i is the predicted probability for class i.

4. Data Collection

This study is based on the data collected by Krajewski et al. (2018) from German highways around the city of Cologne. This analysis refers to the yearlong period between 2017 and 2018. This data set includes 110000 vehicles, which provides for 6000 lane changes in the scope of the observations. Including the data trajectories of all vehicles on a 420-meter stretch of road. Traffic data is recorded frame by frame using a drone positioned above the roadway. During preprocessing, the data is filtered to include vehicles performing lane changes and those nearby. The processed data is then used to train and test two models through one hundred epochs, assessing their performance and improvement.

5. Results and Discussion

Using the datasets collected and processed, this research develops two neural networks that predict lane change maneuvers on the road while being mindful of surrounding vehicles. Only vehicles that conduct lane changes were considered in the training and testing data sets, while those surrounding were included in the social context of the models.

5.1 Numerical Results

Table 1 introduces the best performance points collected from the respective models. While Table 1 shows the CNN model performing better in the data, it does not demonstrate the stability of the RNN model in the present study.

Table 1. Comparative Epoch Performance

Metric	Highest Performing RNN	Highest Performing CNN
Loss	0.6719962282004763	0.4986992165224611
Accuracy	0.727350	0.818873
Miss Rate	0.140381	0.068287
False Alarm Rate	0.073545	0.007959

5.2 Graphical Results

The performance of the developed neural networks has been evaluated using the data sets and the results gathered. It is worth noting in the following figures the stability of the RNN shown and the quality of the models that can produce expected and reliable results.

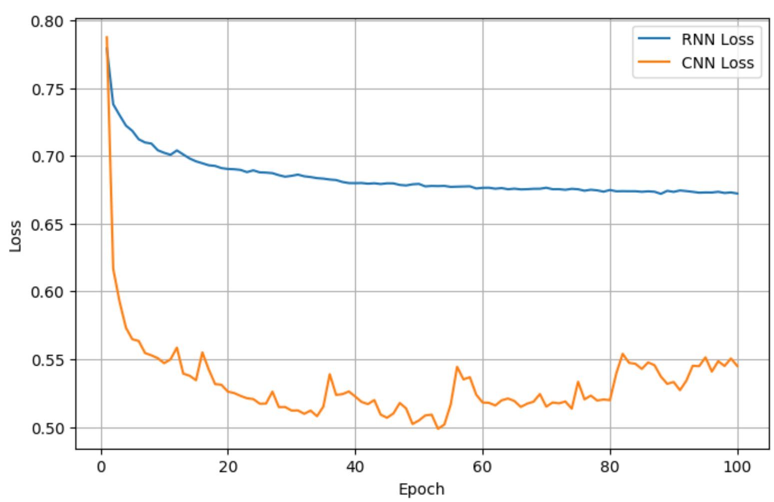

Figure 5. Loss for the models over epochs

Figure 5, presents a loss comparison between the RNN and CNN models over 100 training epochs. The CNN reaches a lower loss but with more fluctuations while the RNN demonstrates a more stable but slower convergence.

Figure 6, illustrates the miss rate of the two models, which corresponds to the number of instances that fail to classify properly. This figure represents the proportion of false negatives out of the total actual positive instances. Minimizing the miss rate is critical to the importance of detecting safe lane change predictions for vehicles.

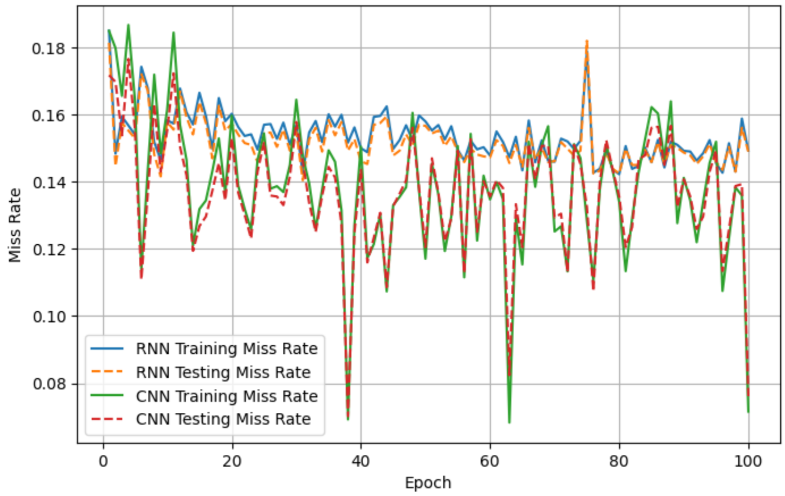

Figure 6. Miss rate for the models with training and testing data over epochs

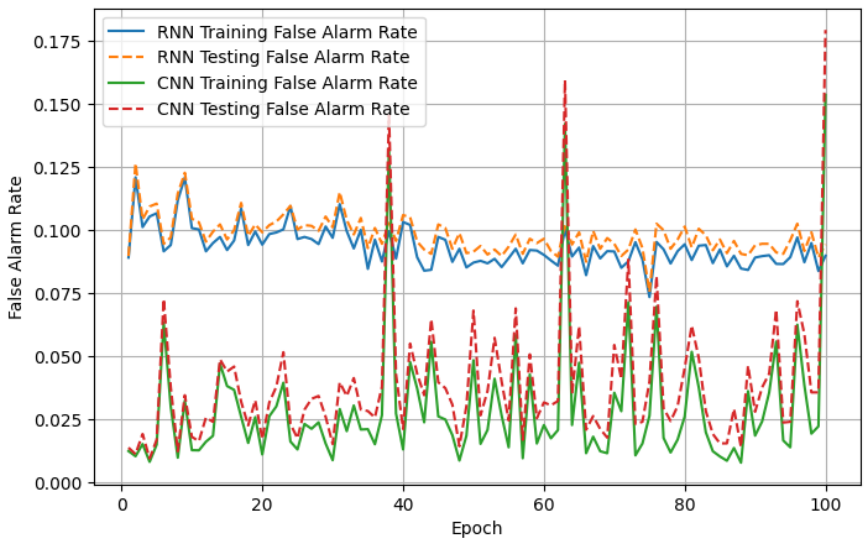

Figure 7. False alarm rate for the models with training and testing data over epochs

On the other hand, Figure 7 measures the opposite of the miss rate by focusing on the proportion of negative instances that are incorrectly classified as positive. The higher this value is the more frequent the models are misclassifying negative instances as positive which is undesirable in the network.

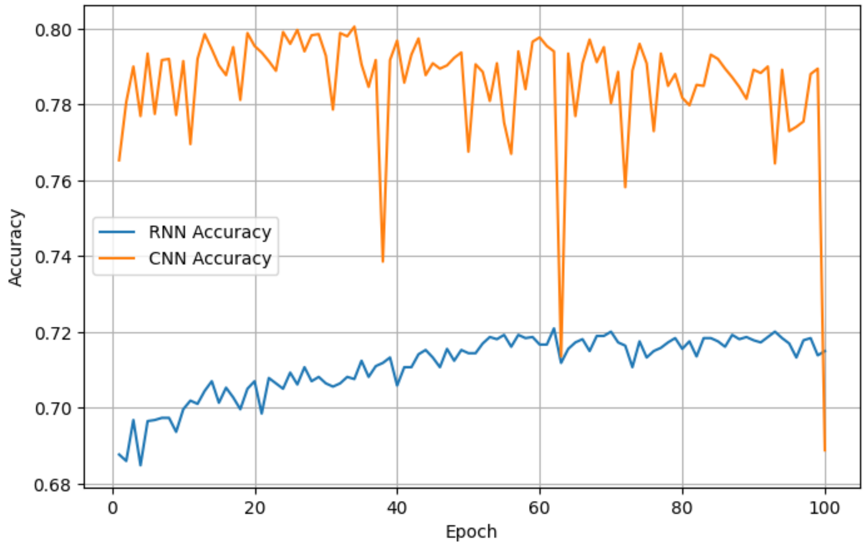

Figure 8. Accuracy for the models with training and testing data over epochs

Figure 8 displays the accuracy of the models for the task. Measuring the correct number of predictions made by the model to see how it is performing overall. Returning a quality value that continuously shows improvement throughout the runtime for the model.

5.3 Proposed Improvements

The models created indicate a starting point that through future work could allow for a reliable and predictable intelligent model for lane change maneuvers. Future steps should work to continue to improve the model with better weights and management of different layers. More data would allow for a higher training size which should allow for an increase in performance and confidence testing. Deep reinforcement learning should be implemented to add a level of depth that the model does not have. More epochs conducted to test the models with more time to converge. The continuation of more iterations to allow for further stabilization of the neural networks would improve results.

5.4 Validation

To validate the performance of the proposed models, we utilized the HighD dataset, which contains data from over 110,00 vehicles. The dataset was split into two subsets: training (used to train the model) and testing (used to evaluate performance), maintaining a consistent 80% training and 20% testing split of the data for both models. To ensure a fair comparison, the models were designed with the same number of layers throughout the process. The validation results, shown in Table 2, indicate that the CNN model outperforms the RNN model in terms of accuracy, achieving an accuracy of 79% compared to the RNN's 71%. The superior performance of the CNN model can be attributed to its ability to extract rich spatial features, which is advantageous for utilizing image-based lane change prediction. However, the RNN model showed better results for stability compared to RNN. With stability being measured as the largest variation in loss between data points after the first ten epochs. At the same time, the CNN required a much longer training time per epoch than the RNN. The improved training efficiency and stability of the RNN can be attributed to its simpler architecture which focuses on time-sequence data without the additional overhead of spatial feature extraction, this results in shorter training times and lower memory consumption.

Table 2. Evaluation Metric on Average

	Accuracy	Stability	Training Time (s)
CNN	0.793439309485329	0.1708057636780187	199.54389421200852
RNN	0.7130205579492859	0.04105825699346455	111.82665116310119

In regard to the computational efficiency of the two models, an analysis is necessary to assess their respective performance in terms of training time and resource utilization as these factors play a crucial role in determining their practical applications for real-time tasks. The RNN model was significantly faster, requiring only 3 hours of training time compared to 5 and a half hours for the CNN model. The RNN model used less memory, making it more computationally efficient for the multiclass classification task. In contrast, the CNN model required greater memory resources due to the complexity of the convolutional layers, which increase the number of parameters and the computational cost of the model for these spatial representations.

To conclude, while both models perform well on different metrics, CNN is more suitable for tasks where accuracy is the primary concern, where the stability of the model and extended training time can be disregarded. While RNN would be better suited for tasks that require a higher stability, faster training, and a quicker convergence.

6. Conclusion

This study provides insight into the distinct types of neural networks and how that could influence traffic applications in a CAV environment. Uniquely highlighting the strengths between these two architectures and how they perform utilizing the same data under different input states. This application shows how CNN and RNN can function while being applied under different approaches in traffic. The observations gathered in these papers allow for improvement upon future research work in these model's applications. This paper highlights the importance of traffic applications and how they can improve efficiency and safety soon. Future research work could be used to help refine and expand upon the understanding of CAV traffic applications in a real-time traffic environment.

References

A. Dongare, R. Kharde, and A. Kachare, "*Introduction to Artificial Neural Network*," Certified *International Journal of Engineering and Innovative Technology (IJEIT)*, vol. 9001, no. 1, pp. 2277–3754, 2008, Available: shttps://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=04d0b6952a4f0c7203577afc9476c2fcab2cba06

A. Mao, M. Mohri, and Y. Zhong, "Cross-Entropy Loss Functions: Theoretical Analysis and Applications." Available: https://proceedings.mlr.press/v202/mao23b/mao23b.pdf

Alessio Micheli, Frank-Michael Schleif, Peter Tiňo (2013). *Novel approaches in machine learning and computational intelligence*. Neurocomputing. 112,1-3.

A. Younesi, M. Ansari, M. Fazli, A. Ejlali, M. Shafique, and J. Henkel, "A Comprehensive Survey of Convolutions in Deep Learning: Applications, Challenges, and Future Trends," arXiv.org, Feb. 28, 2024. https://arxiv.org/abs/2402.15490

A. Younesi, R. Afrouzian, and Y. Seyfari, "A transfer learning ap proach with convolutional neural network for Face Mask Detection," *Journal of Advanced Signal Processing*, vol. 5, no. 1, Jan. 2021, doi: 10.22034/jasp.2022.48447.11

B. Lindemann, T. Müller, H. Vietz, N. Jazdi, and M. Weyrich, "A survey on long short-term memory networks for time series prediction," *Procedia CIRP*, vol. 99, pp. 650–655, 2021, doi: https://doi.org/10.1016/j.procir.2021.03.088.

D. Arpit, B. Kanuparthi, G. Kerg, N. R. Ke, I. Mitliagkas, and Y. Bengio, "h-detach: Modifying the LSTM Gradient Towards Better Optimization," arXiv.org, 2018. https://arxiv.org/abs/1810.03023

F.A. Ford, *Modeling the environment: an introduction to system dynamics models of environmental systems*, Island Press, 1999.

Hochreiter S, Schmidhuber J. 1997. Long short-term memory. *Neural Computation* 9:1735−80
doi: 10.1162/neco.1997.9.8.1735

J. He *et al.*, "Cooperative Connected Autonomous Vehicles (CAV): Research, Applications and Challenges," *2019 IEEE 27th International Conference on Network Protocols (ICNP)*, Oct. 2019, doi: https://doi.org/10.1109/icnp.2019.8888126.

J. Rala Cordeiro, A. Raimundo, O. Postolache, and P. Sebastião, "Neural Architecture Search for 1D CNNs—Different Approaches Tests and Measurements," *Sensors*, vol. 21, no. 23, p. 7990, Nov. 2021, doi: https://doi.org/10.3390/s21237990.

Md. M. Rana and K. Hossain, "Connected and Autonomous Vehicles and Infrastructures: A Literature Review," *International Journal of Pavement Research and Technology*, vol. 16, Nov. 2021, doi: https://doi.org/10.1007/s42947-021-00130-1.

N. Srivastava, G. Hinton, A. Krizhevsky, and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," *Journal of Machine Learning Research*, vol. 15, pp. 1929–1958, 2014, Available: https://www.cs.toronto.edu/~rsalakhu/papers/srivastava14a.pdf

Q. Chen, "F-Cooper: Feature based Cooperative Perception for Autonomous Vehicle Edge Computing System Using 3D Point Clouds," arXiv.org, Sep. 13, 2019. https://arxiv.org/abs/1909.06459

Q. Yang, S. Fu, H. Wang, and H. Fang, "Machine-Learning-Enabled Cooperative Perception for Connected Autonomous Vehicles: Challenges and Opportunities," *IEEE Network*, vol. 35, no. 3, pp. 96–101, May 2021, doi: https://doi.org/10.1109/mnet.011.2000560.

R. Krajewski, J. Bock, L. Kloeker, and L. Eckstein, "The highD Dataset: A Drone Dataset of Naturalistic Vehicle Trajectories on German Highways for Validation of Highly Automated Driving Systems," *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, Nov. 2018, doi: https://doi.org/10.1109/itsc.2018.8569552.

R. Molina-Masegosa, J. Gozalvez and M. Sepulcre, "Comparison of IEEE 802.11p and LTE-V2X: An evaluation with periodic and aperiodic messages of constant and variable size", *IEEE Access*, vol. 8, pp. 121526-121548, 2020.

R. Shoham and H. Permuter, "Amended Cross Entropy Cost: Framework For Explicit Diversity Encouragement," *arXiv.org*, 2020. https://arxiv.org/abs/2007.08140

S. Das, A. Tariq, T. Santos, Sai Sandeep Kantareddy, and I. Banerjee, "Recurrent Neural Networks (RNNs): Architectures, Training Tricks, and Introduction to Influential Research," *Neuromethods*, pp. 117–138, Jan. 2023, doi: https://doi.org/10.1007/978-1-0716-3195-9_4.

World Health Organization, "Global status report on road safety 2023," www.who.int, 2023. https://www.who.int/teams/social-determinants-of-health/safety-and-mobility/global-status-report-on-road-safety-2023

Z. C. Lipton, J. Berkowitz, and C. Elkan, "A Critical Review of Recurrent Neural Networks for Sequence Learning," arXiv.org, 2015. https://arxiv.org/abs/1506.00019

Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, "A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 12, pp. 1–21, 2021, doi: https://doi.org/10.1109/tnnls.2021.3084827.

Biographies

Chase Waelbroeck Graduate student in the Master Program of Electrical and Computer Engineering, Florida Polytechnic University. Bachelor's degree in electrical engineering, Florida Polytechnic University. His research interests are Connected Autonomous Vehicles and Artificial Intelligence.

Rawa Adla, Ph.D. is an Associate Professor of Electrical and Computer Engineering at Florida Polytechnic University. She is a professional in Intelligent Transportation Systems (ITS), bringing together a strong blend of academic and industry experience in the field. Her research interests encompass ITS, autonomous vehicles, connected vehicle technologies (V2V, V2I, V2X), electric vehicles, and battery systems. Her work focuses on developing, analyzing, and testing innovative methodologies aimed at improving traffic safety, enhancing transportation system efficiency, and advancing the performance of electric vehicles and battery technologies.