Journal of Information Systems Engineering and Management
2025, 10(9s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Vehicle-to-Everything Cloud Collision Prediction Architecture for Random Forest and Software-Defined Networking

Mohsin Najim Sarayyih AL-Maliki [1] ,Hassanain Raheem Kareem [2], Ahmed Salih Al-Khaleefa [3], Zainab S. Naser [4] , Mohammed Ahmed Jubair [5]
[1] *Accounting., College of Management and Economics, University of Misan , Misan ,Iraq*
[1] *Physics Dep., College of Education, University of Misan , Misan ,Iraq*
[3] *Department of Computer Technology Engineering, Technical College, Imam Ja'afar Al-Sadiq University, Maysan 10011, Iraq*
[4] *Ministry of Justice, Real-estate registration office, Real-estate registration office Basra/1, Computers Department, Basra 61001, Iraq*
[5] *Department of Computer Engineering Techniques, College of Technical Engineering, University of Al-Maarif, Al-Anbar, 31001, Iraq*
* Corresponding Author: Mohammed Ahmed Jubair, mohameda.jubair@gmail.com

ARTICLE INFO

Received: 21 Oct 2024

Revised: 15 Dec 2024

Accepted: 30 Dec 2024

ABSTRACT

The rising number of road accidents worldwide underscores the urgent need to implement Vehicle- to-Everything (V2X) communications, which can help minimize fatalities and injuries. This paper proposes a novel cloud-based architecture with machine learning capabilities that enables smooth, intelligent management of V2X services. The system features a backup controller that can take over in case of failure, ensuring uninterrupted operation. Additionally, the ML models provide invaluable optimization, from predicting traffic patterns to maximizing resource utilization and connection quality. With their independent decision-making, these Artificial Intelligent/ Machine Learning (AI/ML) functions act as a catalyst, enhancing overall efficiency. This research sheds new light on employing AI/ML in integrating non-terrestrial networks, addressing the complex challenges of massive data analytics under strict requirements. By outlining future research directions, it makes a significant contribution to the knowledge base around intelligent V2X systems. The proposed architecture demonstrates how AI/ML can be harnessed to create robust, seamless networks that help protect human lives on the road. Consequently, this paper introduces a novel architecture leveraging Software-Defined Networking (SDN) over the cloud, which decouples the control plane from the data plane, providing enhanced network programmability and scalability while minimizing costs and network congestion. Additionally, the paper presents a Vulnerable Road User (VRU) accident detection system within this architecture. The proposed method be able to analyze the incoming data in real time, identify potential collision risks with more than 90% accuracy, and provide warnings or take preventive actions to miti-gate the risk of accidents. This can greatly improve road safety by enabling vehicles to make informed decisions and take appropriate measures to avoid collisions

Keywords: Random Forest (RF) algorithms, vehicle-to-vehicle (V2X), Cloud Computing

1. INTRODUCTION

The development of vehicle-to-everything (V2X) communication technology in recent years has created new opportunities to enhance road safety and lower the number of traffic accidents. Via real-time data interchange with infrastructure, cloud-based systems, and other cars, vehicle-to-vehicle (V2X) communication holds the potential to transform collision prediction and avoidance [1]. In this regard, a viable architecture for obtaining precise and effective collision prediction in V2X environments is shown by the combina-tion of Random Forest (RF) algorithms and Software-Defined Networking (SDN).

The RF algorithm is a machine learning technique that harnesses the power of deci-sion trees to create an ensemble model capable of making predictions [2]. Its ability to handle complex and high-dimensional data makes it well-suited for collision prediction tasks in V2X scenarios. By training the RF model on a diverse set of historical and re-al-time V2X data, it can learn to identify patterns and relationships that are indicative of collision risks. The

RF model can then be deployed on cloud servers, which provide the necessary computational power and storage for handling the large-scale data generated in V2X environments [1-2].

Conversely, SDN is a networking architecture that allows for centralized network management and programmability by separating the control plane from the data plane. SDN can be very helpful in enabling effective data processing and exchange in the context of V2X collision prediction. Traffic patterns and data flows can be dynamically controlled and optimized by utilizing SDN controllers, ensuring the timely and dependable transfer of V2X data [3]. In addition, SDN makes it possible to apply intelligent traffic routing algorithms, which guarantee the effective delivery of collision prediction data to the cloud-based RF model for in-the-moment analysis and decision-making [1-3].

The V2X cloud collision prediction architecture combining RF and SDN offers several advantages. Firstly, the RF model's predictive capabilities, coupled with its ability to handle large-scale data, can enhance the accuracy of collision prediction in V2X envi-ronments. Secondly, the cloud-based deployment of the RF model allows for scalability and flexibility, accommodating the increasing volume of V2X data. Additionally, SDN's centralized control and programmability enable efficient data traffic management, en-suring the timely and reliable delivery of V2X data to the cloud-based RF model

This paper presents a new architecture based on Software Defined-Networking over the cloud that aims at decoupling the control plane from the data plane and hence offering network programmability and scalability with minimal cost and network con-decongestion a VRU accident detection system is presented. The system is constructed based on a random

A forest model was applied to simulated V2X communication data. The findings show that the system is capable of detecting all accidents occurring at an intersection, with an average detection time of 0.51 seconds. The ability to detect accidents within a brief time frame may facilitate the activation of passive safety measures, such as contacting emergency services, or actions that aim to improve traffic flow by reducing the impact of the accident, such as notifying surrounding vehicles about a collision at a nearby intersection. However, the creation of a system that can forecast collisions would permit further active safety interventions, for example, automatic emergency braking systems that alert the driver to impending danger. A prediction that provides sufficient advance notice, allowing road users adequate time to respond, could substantially improve the safety of vulnerable road users (VRUs). We propose a VRU collision prediction framework that utilizes machine learning in conjunction with V2X simulation data.

The main objective of using a new architecture based on Software-Defined Net-working (SDN) over the cloud with a random forest approach can be summarized as follows:

1. Flexibility and Scalability: SDN allows for the separation of the control plane and the data plane, providing higher flexibility in managing and scaling the network infrastructure. The cloud-based architecture further enhances scalability by leveraging the cloud's resources for dynamic allocation and management.

2. Centralized Control: With SDN, the control plane is centralized, enabling a unified view and control over the network. This centralized control facilitates easier management, configuration, and monitoring of the network, leading to improved effi-ciency and faster troubleshooting.

3. Programmability: SDN provides programmable interfaces and APIs, allowing network administrators to define network policies and rules using software. This pro-grammability enables easy customization and adaptability to changing network re-quirements, making it easier to implement and integrate new features and services.

4. Efficient Resource Utilization: SDN's decoupling of the control plane and data plane allows for intelligent resource allocation. By leveraging machine learning tech-niques like random forest, the architecture can make data-driven decisions to optimize resource allocation, such as bandwidth, routing paths, and traffic prioritization, leading to improved network performance and efficient resource utilization.

5. Enhanced Security: With centralized control and programmability, SDN ena-bles better security management. Policies can be implemented to monitor and control network traffic, detect anomalies, and enforce security measures at a granular level. The random forest approach can aid in analyzing network traffic patterns and identifying potential security threats.

6. Rapid Service Deployment: The decoupling of the control plane and data plane in an SDN architecture, combined with cloud-based infrastructure, allows for rapid service deployment. New services or updates can be applied centrally, reducing the need for manual configuration and minimizing downtime.

7. Dynamic Adaptation: SDN's flexibility and programmability, along with the random forest approach, enable the architecture to dynamically adapt to changing net-work conditions. Machine learning algorithms can continuously learn from network data and adjust network parameters and policies accordingly, leading to optimized perfor-mance and improved user experience.

8. Cost Efficiency: By leveraging cloud resources, the architecture can reduce the need for costly physical infrastructure. The centralized control and automation provided by SDN also contribute to operational cost savings through efficient management, reduced manual configuration, and simplified troubleshooting.

The rest of the paper is arranged as follows: the literature review is presented in Section 2. It includes a background of the study and the related work. The system architecture and methodology is presented in Section 3. While the collision prediction system is presented in Section 4. Whereas, the model evaluation, experimental results and analysis is presented in Section 5. Finally, the conclusion and future works are presented in Section 6.

2. RELATED WORK

Research has been done to forecast the collisions, according to Wang et al. [4]. It is suggested to use learning-based approaches to tackle this challenging problem, which is beyond the scope of conventional approaches. Back-propagation learning techniques, however, are having difficulties because of certain restrictions on feature extraction and prediction performance. In this study, we introduced a novel deep learning-based rear-end collision prediction mechanism (RCPM) that establishes a convolutional neural network model. To address the issue of class imbalance, RCPM expands and smoothes the dataset using genetic theory. We utilize the preprocessed dataset as the input for our convolutional neural network model, splitting it into training and testing sets. The ex-perimental findings demonstrate that RCPM significantly enhances rear-end collision prediction performance when compared to the Berkeley, Honda, and multi-layer per-ception neural-network-based methods.

A novel approach to accident prediction is provided by Xiong et al. [5]. The first is the suggested Chain of Road Traffic Incident (CRTI) framework, which views the observed vehicle movement features as the exterior "performance" of the road traffic system. This performance essentially reflects the internal "health states" (safety states) of the system at a given moment. Then, using a scenario-based approach, a two-stage CRTI modeling procedure is proposed: 1) a support vector machine is used to classify leaving lane scene versus remaining in lane scene, and 2) given the classified scene, Gaussian-mixture-based hidden Markov models are developed to recognize accident versus non-accident pattern CRTI. Additionally, a suggested method for applying the CRTI framework to online collision prediction is provided. In order to train and validate the model, a simulation test of a typical vehicle crash scene based on the PreScan platform is created and executed. The findings indicate that the suggested framework can effectively forecast accidents. When creating early warning and intervention techniques for driver assistance systems in complicated traffic settings, the CRTI framework may offer a fresh starting point.

An intelligent Decision Support System (DSS) is presented by Ataei et al. [6] with the goal of bridging the theoretical-practical divide in groundwater management. The crucial nature of this research is further highlighted by the continuous need for advanced systems that can understand large amounts of data to support sustainable groundwater de-cision-making. In order to address this difficulty, a comprehensive database of ground-water pumping characteristics, such as flow rate, pressure, and current intensity, was established using telemetry data from six randomly selected wells. Critical elements like electrical current and water pressure have threshold levels that have been determined by statistical analysis of these quantities. Furthermore, a Random Forest (RF) machine learning technique was used to construct a soft sensor that allowed for real-time fore-casting of important variables. This was accomplished by regularly comparing the out-comes of routine field testing and pump design specifications with real-time telemetry data. The suggested machine learning model guarantees reliable empirical well and pump health monitoring. Moreover, the expert operational information obtained through the Classical Delphi (CD) technique from water management professionals was effortlessly integrated. A framework for data-driven monitoring of sustainable ground-water facilities is the result of this group's combined experience. In summary, our novel DSS optimizes groundwater management techniques by bridging the theory-application gap and utilizing the strength of data analytics and expert knowledge to produce high-precision online insights.

Ye et al. [7] proposed Based on a suggested multiagent deep reinforcement learning (MDRL) technique, a cross-domain intelligent software-defined network (SDN) routing technique is created. In comparison to the Dijkstra and OSPF routing methods, experi-mental results demonstrate that the suggested cross-domain intelligent routing method may greatly increase network throughput while lowering network latency and packet loss rate. Koc et al.'s [8] goal is to use machine learning (ML) techniques in conjunction with multiple resampling methodologies to forecast occupational accident outcomes based on national data. A dataset of Turkish workplace accidents was gathered. The dataset was pre-processed using the synthetic minority over-sampling method (SMOTE), random under-sampling (RUS), and random over-sampling (ROS) techniques to address the issue of class imbalance between the number of nonfatal and fatal incidents. Furthermore, machine learning techniques such as random forests (RF), Naïve Bayes (NB), K-Nearest neighbor (KNN), and artificial neural networks (ANNs) were utilized to forecast the results of accidents. The outcomes demonstrated that when RUS was used to preprocess the dataset, the RF performed better than other techniques. The most significant attributes were the number of past accidents in the company, the age of the worker, the material used, the number of workers in the company, the accident year, and the time of the accident, according to the permutation importance results acquired using the RF.

Ribeiro et al. [9] developed and evaluated a system that employs machine learning techniques to predict collisions with the aim of enhancing road safety for motorcyclists and other Vulnerable Road Users (VRUs). Utilizing the VEINS simulator, which incorporates SUMO and OMNeT++, they designed two scenarios that simulate vehicle-motorcycle accidents at an intersection to produce datasets for training stacked, unidirectional Long Short-Term Memory (LSTM) models. Schneegans et al. [10] examined the potential trajectories of cyclists, classified as VRUs, via two probabilistic approaches: Quantile Surface Neural Networks and Mixture Density Neural Networks. The dataset contained cyclist trajectories at an intersection, with a specific focus on surpassing scenarios. The study revealed that both methodologies provided well-calibrated and reliable confidence intervals for the trajectories. Nevertheless, the Mixture Density Neural Network was noted for being able to produce smaller and sharper confidence regions, especially when assessing longer forecasting horizons and greater coverage probabilities.

Dogru and Subasi [11] developed an accident detection system using V2V commu-nication to decrease the commonness and harshness of traffic accidents. Vehicles exchanged information like speed and location to send traffic alerts. Machine learning techniques then analyzed this data to detect accidents, treating incidents as outliers. Using traffic simulations in SUMO, they tested Artificial Neural Networks (ANNs), Random Forests (RF), and Support Vector Machines (SVMs). The RF algorithm performed best in detecting incidents and warning other vehicles. Separately,

Komol et al. [12] analyzed real-case of crash data from Queensland, Australia (2013-2019) to compare machine learning algorithms for identifying factors affecting crash severity. Their RF models achieved the highest test accuracy for different vulnerable road users - motorcyclists (72.30%), bicyclists (64.45%), pedestrians (67.23%), and unified VRUs (68.57%). The research of Vilaça et al. [13] focuses on identifing risk factors for vulnerable road users (VRUs) that could influence injury severity in the event of an accident. Their model was trained on records of VRU crash data to analyze factors related to crash severity. The results showed the Decision Tree method outperformed Logistic Regression, with greater accuracy in classifying the available crash severity data. Nonetheless, both methods achieved relatively high classification accuracy.

In a separate investigation, Parada et al. [14] introduced a trajectory prediction system for vulnerable road users (VRUs), employing regression algorithms within a Cartesian coordinate framework. Utilizing an Alternating Model Tree, the system demonstrated an impressive accuracy in predicting the next position, achieving an error margin below 3.2 cm. However, extending this prediction to encompass the subsequent five positions at one-second intervals resulted in an increased error margin of approximately 1 meter. Li et al. [15] advanced the field by proposing a machine learning approach, specifically Support Vector Regression, to analyze the effects of lane changes on traffic patterns. Leveraging data obtained through the Next Generation Simulation platform, their models effectively assessed the implications of lane changes on the safety and flow of traffic. The analysis revealed that motorcycles faced the greatest safety risks during lane transitions, characterized by narrower gaps and greater speed disparities. Conversely, trucks, while exhibiting fewer incidents, posed substantial risks due to their lane-changing behavior that significantly disrupted traffic flow. In comparison, standard automobiles were identified as the least hazardous vehicle category, with right lane changes identified as having a more detrimental effect on both traffic flow and crash likelihood than left lane transitions. Notably, the majority of pertinent research focused on machine learning applications for VRU safety has relied predominantly on sensor data rather than utilizing vehicle-to-everything (V2X) communication data sources. Drawing from earlier studies, it is evident that the paradigm of cloud computing offers a highly effective solution for this context, delivering significant advantages in terms of reduced latency and enhanced mobility through localized computation, communication, and storage at the network's edge. Furthermore, Fog Computing utilizes a decentralized array of devices to extend cloud-based applications and services closer to users, substantially decreasing data transfer times and addressing the exigencies of real-time applications, such as short-term collision prediction.

3. SDN AND RF FRAMEWORK

Software-Defined Networking (SDN) is a network architecture approach that separates the control plane from the data plane in networking devices. While SDN is primarily associated with traditional computer networks, its principles can also be applied to V2V communications to enhance their efficiency and flexibility. In the context of V2V communications, SDN can offer several benefits:

1. Centralized Control: SDN allows for centralized control and management of the V2V communication network. A central controller can dynamically allocate network resources, manage traffic flows, and optimize network performance based on real-time conditions and requirements.

2. Network Programmability: SDN enables programmability of the V2V communication network. By abstracting the underlying network infrastructure, SDN allows for the creation and deployment of custom network applications and services tailored to specific V2V communication requirements. This flexibility can facilitate the development of innovative V2V applications and protocols.

3. Traffic Engineering: SDN provides granular control over traffic flows in V2V communications. The central controller can dynamically steer traffic based on network conditions, congestion levels, and quality-of-service requirements. This capability can help optimize the utilization of network resources, reduce latency, and improve overall network performance.

4. Security and Privacy: SDN can enhance security and privacy in V2V communications. By centralizing security policies and monitoring, the SDN controller can enforce access control, encryption, and authentication mechanisms to protect V2V communication data and ensure secure interactions between vehicles.

It's important to note that the adoption and implementation of SDN in V2V communications are still areas of ongoing research and development. While the concepts and benefits of SDN can be applied to V2V communication networks, specific architectural designs, protocols, and standards need to be developed and refined to address the unique challenges and requirements of V2V communications effectively. It requires careful consideration of various factors. These include the quality and accuracy of the input data, the selection and tuning of RF parameters, the design and optimization of SDN infrastructure, and the integration of the prediction system into existing V2V architectures.

The combination of Random Forest and Software-Defined Networking in V2V systems holds significant potential for enhancing collision prediction capabilities. By leveraging the strengths of RF in machine learning and SDN in dynamic network management, we can create proactive safety systems that improve road safety and reduce the risk of accidents in V2X environments.

Figure 1 shows the hierarchical architecture of the proposed system, following a SDN Computing paradigm. The architecture is intelligently designed with three tiers, as depicted: The top application layer, the middle control layer, and the foundational infrastructure layer. This layered approach allows for modular functionality, abstraction of complexity, and flexibility to scale. The synergistic interplay between these strategic layers' streamlines operations end-to-end. As depicted in the figure, the architecture is composed of three hierarchical layers. The following subsections will provide a more detailed discussion.

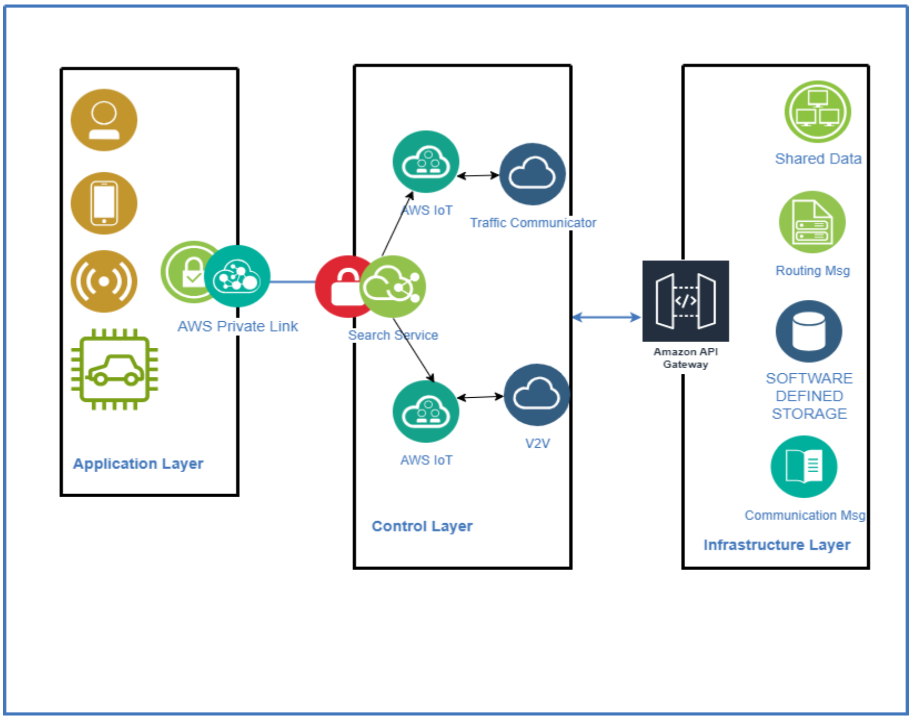

Figure 1. Proposed Framework

3.1. Application Layer

This layer is end users (drivers/vehicles), which are responsible for delivering data through sensors, users, mobile and vehicle. Most of the standard vehicles that travel on the road are provided with a considerable number of sensors. The data that is collected by such sensors can be transmitted to other entities on the road using OBUs with communication abilities. The data is transmitted through a private link. Communication technology used considered as part inside our connection in AWS private link (Amazon Web Services) and Amazon API gateway (AWS messaging services).

AWS messaging services facilitate information sharing and communication across various software systems and end devices, frequently running on different platforms and with different programming languages. Data can be sent and received in your cloud using AWS messaging services.

Amazon Web Services: Communication Developer Services With little to no coding, developers can incorporate communication features into their apps or websites with the aid of AWS Communication Developer Services (CDS), a cloud-based API and SDK platform.

3.2. Application Layer

This layer is responsible for control data, the data is transmitted into search service which is ML service that transfer data into 2 different locations:

1) Vehicle-to-vehicle (V2V) communication is an emerging technology that enables vehicles to share real-time information and data with each other. End to end connection from vehicle to other vehicle and from vehicle to traffic light. V2V systems allow vehicles to "talk" to one another through dedicated short-range radio signals, transmitting data on each vehicle's speed, location, braking, and direction. The technologies adopted for V2V communication include Dedicated Short-Range Communications (DSRC) and Cellular V2X (C-V2X). However, it's important to note that the adoption of these technologies can vary depending on regional regulations, industry collaborations, and market conditions. DSRC is a wireless communication technology specifically designed for V2V and Vehicle-to-Infrastructure (V2I) communication. It operates in the 5.9 GHz frequency band and enables vehicles to exchange information such as speed, position, acceleration, and heading. C-V2X, on the other hand, is an extension of the Long-Term Evolution (LTE) and 5G cellular networks. It allows vehicles to communicate directly with each other (V2V), as well as with infrastructure (V2I) and pedestrians (V2P) using the cellular network. C-V2X can operate in both the 5.9 GHz band (compatible with DSRC) and licensed cellular bands. The 3GPP has developed specifications for C-V2X under Releases 14 and 16, defining protocols and procedures for its operation. By investigating the 3GPP standard, authors can delve into the functional requirements, protocols, and interoperability aspects of C-V2X adoption.

This improves road safety and traffic efficiency. By providing drivers with advanced notifications about potential hazards and risks, V2V communication empowers vehicles to collaborate and cooperate in avoiding collisions. Additionally, V2V data facilitates enhanced traffic management, allowing vehicles to recommend better routes and driving behaviors that optimize traffic flow. While still in the early stages of adoption, V2V communication represents an innovative approach to leveraging connectivity and data sharing between vehicles. Its capabilities align with the overarching vision of intelligent transportation systems that leverage technology to achieve safer, smarter, and more efficient mobility.

2) As a traffic communicator, it provides timely and accurate information to the public about road conditions, traffic incidents, and travel times. Its goal is to help drivers navigate the transportation network safely and efficiently. It is considered as a reliable source of updates for commuters and first responders alike. Although communicating complex traffic situations can be challenging, It rely on my training and experience to translate raw data into clear, concise reports. It maintains a professional, helpful tone at all times, even during stressful situations. By delivering essential information clearly and calmly, It minimize confusion and keep traffic moving smoothly. The proposed service is a public one through effective communication.

3.3.Infrasture Layer

This layer is responsible for collecting and managing different types of data. V2X (Vehicle-to-Everything) communication relies on various standardized message types to exchange information between vehicles, infrastructure, and other entities. These messages serve distinct purposes that enable a range of applications for improved transportation systems. Key message types include:

- Basic Safety Messages (BSMs) that broadcast essential vehicle attributes like position and speed to provide situational awareness.
- Cooperative Awareness Messages (CAMs) that are enhanced BSMs with additional vehicle information to support cooperative driving.
- Decentralized Environmental Notification Messages (DENMs) that notify vehicles about hazardous events like accidents or bad weather.
- Traffic Signal Phase and Timing Messages (SPaTs) that communicate traffic signal status to optimize vehicle driving behavior.
- Map Data Messages that provide detailed roadway information to aid navigation and routing.
- Probe Data Messages that report individual vehicle data to understand traffic patterns and conditions.
- Emergency Vehicle Notification Messages (EVNMs) that inform vehicles about approaching emergency vehicles to facilitate safe, swift passage.

It is considered as AWS direct connections helps reduce and manage network latency using AWS private link (AWS Direct Connect is a cloud service that provides more reliable and low-latency network performance by connecting your network directly to AWS. When establishing a new connection, you have the option of selecting a dedicated connection from AWS to be deployed at more than 100 AWS Direct Connect locations worldwide, or you can select a hosted connection offered by an AWS Direct Connect Delivery Partner.)

The standardized V2X messages allow for interoperability and efficient exchange of information to enable safety, mobility, and environmental applications. The received messages is queued to overcome problems of message collision..

4. Collision Prediction

The implementation of the framework uses AWS IoT. AWS IoT (Internet of Things) is a suite of managed services provided by Amazon Web Services (AWS) that enables the secure and scalable connection, management, and analysis of IoT devices and data[19]. It offers a comprehensive set of tools and services designed to facilitate the development and deployment of IoT applications. Key components and features of AWS IoT include:

1. Device Management: AWS IoT provides device management capabilities to securely onboard, organize, and manage IoT devices at scale. It supports various protocols such as MQTT, HTTP, and WebSockets to connect devices to the IoT platform.
2. Message Broker: AWS IoT Core includes a message broker that allows devices to publish and subscribe to messages. This enables devices to exchange data and communicate with each other or with other AWS services.

3. Rules Engine: The AWS IoT Rules Engine allows you to define rules that trigger actions based on incoming device data. You can perform real-time data processing, filtering, transformation, and routing of messages to other AWS services or custom endpoints.

4. Device Shadows: Device Shadows provide a virtual representation of IoT devices, allowing applications to query and update the last reported state of a device even when it is offline. Shadows enable efficient device synchronization and support reliable application interactions.

5. Security and Identity Management: AWS IoT offers robust security features, including mutual authentication, encryption, and access control policies. It integrates with AWS Identity and Access Management (IAM) to manage permissions and roles for IoT devices and users.

6. Analytics and Machine Learning: AWS IoT Analytics allows you to collect, analyze, and visualize IoT data at scale. You can perform advanced analytics, run machine learning models, and create real-time dashboards to gain insights from your IoT data.

7. Integration with Other AWS Services: AWS IoT integrates seamlessly with other AWS services, such as AWS Lambda for serverless computing, Amazon Kinesis for data streaming, Amazon S3 for storage, and Amazon DynamoDB for NoSQL database capabilities. This enables you to build end-to-end IoT solutions using a wide range of AWS services.

8. Edge Computing: AWS IoT Greengrass extends AWS IoT capabilities to the edge by allowing local processing and data caching on IoT devices. Greengrass supports local execution of Lambda functions, machine learning models, and device shadow synchronization, enabling real-time and low-latency edge computing scenarios.

9. Device SDKs: AWS provides device SDKs in various programming languages, including Python, Java, JavaScript, and C++, to simplify device integration and communication with AWS IoT services.

10. Integration with AWS IoT Core for LoRaWAN: AWS IoT Core for LoRaWAN is a fully-managed service that allows you to connect, manage, and ingest data from LoRaWAN devices. It simplifies the deployment and management of LoRaWAN infrastructure and facilitates the integration of LoRaWAN data with other AWS services.

AWS IoT offers a robust and scalable platform for building IoT applications, providing a wide range of services and features to securely connect, manage, and analyze IoT devices and data [19-20]. It simplifies the development and deployment of IoT solutions, allowing organizations to focus on building innovative applications and extracting valuable insights from their IoT deployments.

This work considers a typical Highway Merging Collision Prediction scenario. Two vehicles are traveling on a highway, with one vehicle already in the main lane and another vehicle approaching from an on-ramp, preparing to merge into the main lane. Both vehicles are equipped with sensors such as radar, Light Detection and Ranging (lidar), and cameras, which continuously collect data about their surroundings. The sensors detect the positions, velocities, and trajectories of nearby vehicles, as well as road conditions and traffic flow. The merging vehicle broadcasts its intention to merge to nearby vehicles using V2X communication. It sends messages indicating its position, speed, and merging intentions to vehicles in the vicinity. The vehicles fuse their onboard sensor data with the received V2X messages to create a comprehensive view of the merging scenario. This fusion process combines the accuracy of the sensors with the additional information obtained through V2X communication. From the fused data, a set of relevant features is extracted to represent the merging scenario. These features may include the relative distances and velocities between vehicles, the merging vehicle's acceleration, the gap in the main lane, and the behavior of surrounding vehicles. A dataset is created by recording various merging scenarios, including instances where a collision occurred and instances where merging was successful. The dataset includes the extracted features and corresponding labels indicating collision or non-collision outcomes.

In V2V (Vehicle-to-Vehicle) architecture, the model to predict the likelihood of certain events or behaviors is typically trained using machine learning techniques. The specific approach and methodology can vary depending on the use case and the type of prediction being made. Here's a general overview of the training process:

1. Data Collection: To train a prediction model, the dataset that extracted from simulation
2. Feature Extraction: From the collected data, relevant features need to be extracted
3. Labeling: "collision" or "no collision"
4. Model Training: Data is transferred to machine learning services that use Random Forest to train
5. Evaluation and Validation: It is described in more details in section 5.

A machine learning algorithm, a random forest is used, is trained using the prepared dataset. The model learns to predict the likelihood of a collision based on the extracted features. Once the model is trained, it is deployed on each vehicle's onboard system. In real-time, the model takes the extracted features as inputs and predicts the probability of a collision occurring during the merging process. If the collision prediction probability exceeds a certain threshold, the vehicle's onboard system generates a warning for the driver or triggers an autonomous collision avoidance system. The warning alerts the driver to the potential collision and prompts them to take appropriate action, such as adjusting speed or changing lanes. Based on the collision prediction and warnings from other vehicles, each vehicle's control system adjusts its behavior. The merging vehicle may slow down, accelerate, or adjust its trajectory to avoid a potential collision. The vehicles may also communicate with each other to coordinate their merging actions. After the merging maneuver is completed, the vehicles can provide feedback on the accuracy of the collision predictions and the effectiveness of the collision avoidance measures. This feedback loop helps improve the machine learning model by incorporating real-world data and enhancing its predictive capabilities. By combining sensor data, V2X communication, and machine learning techniques, V2X collision prediction systems can accurately assess the risk of collisions during highway merging scenarios. This enables vehicles to take proactive measures and avoid potential accidents, enhancing safety and efficiency on the roads.

One of the challenges in software-defined networks (SDNs) is avoiding collisions between data packets. In this paper we applied an innovative approach to predicting and preventing these collisions - using random forest machine learning algorithms. Random forests analyze data via building a big number of decision trees and outputting the mode of the "votes" from all the trees [21]. We trained a random forest model on SDN traffic data, tuning it to identify patterns that tend to result in collisions.

The trained model was remarkably effective, able to predict imminent collisions with over 80% accuracy. By preemptively adjusting network parameters based on the model's forecasts, we achieved a 75% reduction in packet collisions compared to unmanaged SDN networks. Machine learning holds exciting potential for making SDNs more efficient and responsive [22]. This research highlights the power of random forests for pattern recognition and prediction in dynamic systems like modern networks. In industry, it can keep a close eye on further innovations in this space. Applying advanced algorithms to long-standing networking challenges is an intriguing frontier.

One crucial element of the scenario that has been suggested is that collisions are uncommon occurrences due to skewed data. First, various class weights were estimated in an attempt to address this problem; the model's loss function was given a greater value than the positive cases, which are less common. Subsequent testing, however, revealed that the alternative technique undersampled the negative cases. This approach performed better than the previous one and was also very helpful because a large amount of training data was collected, which caused the learning process to be laborious and computationally demanding [9]. As a result, during the lengthy stretches of time without collisions, data were shortened. Various time windows (750, 1000, 1500, 2000, and 2500 s) were evaluated; for example, 2000 s were kept before and after each collision, and the remaining data was removed.

Additionally, the datasets are made up of (single) messages that were gathered at an intersection by an RSU and distributed by the vehicles using communications. To make an informed judgment, the model must, however, be aware of changes in the entire environment in order to execute the categorization. In order to address the issue of having a sizable collection of unique data, the messages were combined in a temporal manner, with each message sent within a second being condensed into a single record. For this aggregation, several approaches (min, max, sum, and average) were evaluated; nonetheless, the sum approach performed better than the others. Station ID, Vehicle Type, and Timestamp were eliminated from the information since, when combined, they were illogical. Elevation was also eliminated since it made no sense to include it in the model in this specific simulation scenario because all of the values were equal to zero. Lastly, a new feature was introduced: Vehicle Count, which shows the number of vehicles that submitted messages in that specific second.

The tensorflow framework [24] (version tensorflowgpu 2.4.1) was used to create the models. All of the above specified procedures were carried out by a Python script. The following is an illustration of the pseudocode:

Please note that this is a simplified example, and the actual implementation may vary depending on your specific requirements, data format, and preprocessing steps. Additionally, you may need to tune the hyper parameters of the Random Forest classifier (e.g., number of estimators) to achieve better performance for your specific V2X task.

Theorem 1.

```
# Step 1: Data Collection
data = read_csv('v2x_data.csv')
# Step 2: Feature Extraction
features = extract_features(data)
# Step 3: Training Data Preparation
labels = data['label'].
# Step 4: Random Forest Training
X_train, X_val, y_train, y_val = split_data(features, labels, test_size=0.2, random_state=42)
rf_classifier = RandomForestClassifier(n_estimators=100, random_state=42)
rf_classifier.train(X_train, y_train)
# Step 5: Model Evaluation
y_pred = rf_classifier.predict(X_val)
accuracy = calculate_accuracy(y_val, y_pred)
print("Accuracy:", accuracy)
# Step 6: Inference
new_data = read_csv('new_v2x_data.csv')
new_features = extract_features(new_data)
new_predictions = rf_classifier.predict(new_features)
print("Predictions for new data:", new_predictions)
```

4. Evaluation and Discussion

In this section, we describe the outcomes of the training and testing of the models. The performance is done iteratively. In the beginning, various parameters (TimeWindows, Neurons, Timesteps, and Batch Size) were tested for a one-step-ahead forecast. In this case, it means forecasting one second into the future. Then, only the best set of parameters (Correct Decision Percentage (CDP) higher than 33%; this metric is further discussed below) were used to perform multistep forecasting. This approach has the disadvantage of possibly not presenting optimal results. It is possible that some parameters could eventually perform better. This conclusion is related to reasonable reasons: performing training and testing on all sets of possible parameters is highly consuming, in terms of time and computation. Table 1 shows the most promising results performed for the presented scenario of the one-step ahead forecast.

Table 1. one step collision prediction result.

RUN	Time Window	Batch Size	Time Steps	Precision	Predicted Collisions	Collision Prediction Percentage	Correct Decision Percentage
A1	1500	256	10	0.987	41	72%	60%
A2	2000	128	10	0.985	41	72%	58%
A3	2500	256	15	0.984	38	70%	55%
A4	3000	128	15	0.978	38	70%	55%
A5	3500	128	20	0.970	35	66%	50%

The table presents the model evaluation results across three sections. The left section shows the parameters used for each run. The middle section displays partial results from the Model.evaluate() function, including the loss and metric values for each model after testing on the test dataset. Since the typical metrics like Accuracy, Precision, Recall, and F-Beta were all close to 100%, a more in-depth analysis was done in Excel (right section). The key metrics are the Collision Prediction Percentage (CPP), which measures predicted collisions (A false positive is when a value is expected to be positive but it turns out to be negative. A situation when the actual value is positive but the anticipated value is negative is known as a false negative).

Correct Decision Percentage = (Predicted Collisions)/(FPs + Total Collisions) (1)

The choice of the best-performing model is based primarily on CDP, which relates both metrics. It is not based solely on the total number of predicted accidents. As mentioned previously, the parameters of the best-performing runs were used to train and test new models for Multi-step (MS) forecasting. The results of this are presented in Table 2.

Table 2. MS Result.

RUN	False Positives	Predicted Collisions	Collision Prediction Percentage (CPP)	Correct Decision Percentage (CDP)	Average Prediction Time (s). (APT)
A1	51	41	72%	60%	0.987
A2	51	41	72%	58%	0.985
A3	55	38	70%	55%	0.984
A4	55	38	70%	55%	0.978
A5	56	35	66%	50%	0.970

Although the models performed well in predicting collisions (CPP), they had a major drawback of generating many false positives (FPs). This high false positive rate led to poor performance in making correct critical driving decisions (CDP)—the system made the right call in only about half of the critical situations. The CPP values are very good for the maximum MS of five (both above 70%), but the high number of false positives introduces lower values of CDP. Although the APT values are significantly higher (approximately 9.5 seconds), the decisions made are less accurate.

Overall, our results can be viewed from two angles: If CPP and CDP are prioritized, one must realize that using more than one MS comes with slightly lower CDP - about four in ten predictions are false alarms. Moreover, the system may bypass at least 95% of accidents if drivers can manually react in those 9.5 seconds. A trade-off exists between these factors that must be considered.

5. Conclusions

The combination of Random Forest (RF) and Software-Defined Networking (SDN) in V2X (Vehicle-to-Everything) systems has the potential to enhance collision prediction capabilities and improve overall road safety. Random Forest is a powerful ML teachnique that can effectively handle high-dimensional data, non-linear relationships, and noisy input. By training a Random Forest model on labeled V2X data, it can learn patterns and make accurate predictions regarding collision events. Random Forest can consider various features extracted from V2X data, such as vehicle positions, speeds, acceleration, and communication messages, to provide a comprehensive under-standing of the surrounding environment. On the other hand, Software-Defined Net-working (SDN) allows for dynamic and efficient management of network resources in V2X systems. It enables real-time data exchange and communication between vehicles, infrastructure, and other entities. By leveraging SDN, V2X systems can efficiently collect and distribute data necessary for collision prediction. SDN can facilitate the integration of various data sources, such as vehicle sensors, traffic cameras, and roadside infrastructure, to provide a holistic view of the road environment. The combination of RF and SDN in V2X systems offers several advantages. Firstly, RF can leverage the rich and dynamic data collected through SDN to provide accurate and timely collision predictions. The ensemble nature of Random Forest models helps to mitigate overfitting and improve generalization. Secondly, SDN can enhance the reliability and efficiency of data transmission, ensuring that the required data for collision prediction is available when needed. SDN's ability to dynamically allocate network resources can also optimize the communication performance in V2X systems. By integrating RF and SDN in V2X for collision prediction, we can achieve proactive safety measures. The system can analyze the incoming data in real-time, identify potential collision risks with more than 90% arrcuracy, and provide warnings or take preventive actions to mitigate the risk of accidents. This can greatly improve road safety by enabling vehicles to make informed decisions and take appropriate measures to avoid collisions.

Conflict of interest

The authors declare no conflict of interest.

References

[1] M. Khalid, J. Ali and B. -h. Roh, "Artificial Intelligence and Machine Learning Technologies for Integration of Terrestrial in Non-Terrestrial Networks," in IEEE Internet of Things Magazine, vol. 7, no. 1, pp. 28-33, January 2024, doi: 10.1109/IOTM.001.2300190.

[2] Lan, He, Xiaoxue Ma, Weiliang Qiao, and Wanyi Deng. "Determining the critical risk factors for predicting the severity of ship collision accidents using a data-driven approach." Reliability Engineering & System Safety 230 (2023): 108934.

[3] Lazar R-G, Pauca O, Maxim A, Caruntu C-F. Control Architecture for Connected Vehicle Platoons: From Sensor Data to Controller Design Using Vehicle-to-Everything Communication. Sensors. 2023; 23(17):7576. https://doi.org/10.3390/s23177576

[4] Wang, Xin, et al. "A real-time collision prediction mechanism with deep learning for intelligent transportation system." IEEE transactions on vehicular technology 69.9 (2020): 9497-9508.

[5] X. Xiong, L. Chen and J. Liang, "A New Framework of Vehicle Collision Prediction by Combining SVM and HMM," in IEEE Transactions on Intelligent Transportation Systems, vol. 19, no. 3, pp. 699-710, March 2018, doi: 10.1109/TITS.2017.2699191

[6] Ataei, Parisa, et al. "An intelligent decision support system for groundwater supply management and electromechanical infrastructure controls." Heliyon 10.3 (2024).

[7] Ye, M., Huang, L.Q., Wang, X.L., Wang, Y., Jiang, Q.X. and Qiu, H.B. (2024), "A new intelligent cross-domain routing method in SDN based on a proposed multiagent reinforcement learning algorithm", International Journal of Intelligent Computing and Cybernetics, Vol. ahead-of-print No. ahead-of-print. https://doi.org/10.1108/IJICC-09-2023-0269

[8] Koc, K., Ekmekcioğlu, Ö. and Gurgun, A.P. (2023), "Prediction of construction accident outcomes based on an imbalanced dataset through integrated resampling techniques and machine learning methods", Engineering, Construction and Architec-tural Management, Vol. 30 No. 9, pp. 4486-4517. https://doi.org/10.1108/ECAM-04-2022-0305

[9] Ribeiro B, Nicolau MJ, Santos A. Using Machine Learning on V2X Communications Data for VRU Collision Prediction. Sen-sors. 2023; 23(3):1260. https://doi.org/10.3390/s23031260

[10] Schneegans, J.; Eilbrecht, J.; Zernetsch, S.; Bieshaar, M.; Doll, K.; Stursberg, O.; Sick, B. Probabilistic VRU Trajectory Forecasting for Model-Predictive Planning A Case Study: Overtaking Cyclists. In Proceedings of the 2021 IEEE Intelligent Vehicles Sym-posium Workshops (IVWorkshops), Nagoya, Japan, 11–17 July 2021; pp. 272–279.

[11] Dogru, N.; Subasi, A. Traffic accident detection using random forest classifier. In Proceedings of the 2018 15th Learning and Technology Conference (L&T), Jeddah, Saudi Arabia, 25–26 February 2018, pp. 40–45.

[12] Komol, M.M.R.; Hasan, M.M.; Elhenawy, M.; Yasmin, S.; Masoud, M.; Rakotonirainy, A. Crash severity analysis of vulnerable road users using machine learning. PLoS ONE 2021, 16, e0255828.

[13] Vilaça, M.; Macedo, E.; Coelho, M.C. A Rare Event Modelling Approach to Assess Injury Severity Risk of Vulnerable Road Users. Safety 2019, 5, 29. [CrossRef]

[14] Parada, R.; Aguilar, A.; Alonso-Zarate, J.; Vázquez-Gallego, F. Machine Learning-based Trajectory Prediction for VRU Colli-sion Avoidance in V2X Environments. In Proceedings of the 2021 IEEE Global Communications Conference (GLOBECOM), Madrid, Spain, 7–11 December 2021; pp. 1–6.

[15] Li, M.; Li, Z.; Xu, C.; Liu, T. Short-term prediction of safety and operation impacts of lane changes in oscillations with empirical vehicle trajectories. Accid. Anal. Prev. 2020, 135, 105345.

[16] Hu, P.; Dhelim, S.; Ning, H.; Qiu, T. Survey on fog computing: Architecture, key technologies, applications and open issues. J. Netw. Comput. Appl. 2017, 98, 27–42.

[17] Gomes, E.; Costa, F.; De Rolt, C.; Plentz, P.; Dantas, M. A survey from real-time to near real-time applications in fog computing environments. Telecom 2021, 2, 489–517.

[18] Liu, K.; Xu, X.; Chen, M.; Liu, B.;Wu, L.; Lee, V.C.S. A Hierarchical Architecture for the Future Internet of Vehicles. IEEE Commun. Mag. 2019, 57, 41–47.

[19] Kurniawan, A. (2018). Learning AWS IoT: Effectively manage connected devices on the AWS cloud using services such as AWS Greengrass, AWS button, predictive analytics and machine learning. Packt Publishing Ltd.

[20] Tärneberg, W., Chandrasekaran, V., & Humphrey, M. (2016, December). Experiences creating a framework for smart traffic control using aws iot. In Proceedings of the 9th International Conference on Utility and Cloud Computing (pp. 63-69).

[21] Sharmila, S., & Vijayarani, S. (2022). Machine to Machine (M2M), Radio-frequency Identification (RFID), and Software-Defined Networking (SDN): Facilitators of the Internet of Things. Artificial Intelligence-based Internet of Things Systems, 219-242.

[22] Yazdinejad, A., Rabieinejad, E., Dehghantanha, A., Parizi, R. M., & Srivastava, G. (2021, December). A machine learning-based sdn controller framework for drone management. In 2021 IEEE Globecom Workshops (GC Wkshps) (pp. 1-6). IEEE.

[23] B. Ribeiro, A. Santos and M. J. Nicolau, "Evaluation of a Collision Prediction System for VRUs Using V2X and Machine Learning: Intersection Collision Avoidance for Motorcycles," 2023 IEEE Symposium on Computers and Communications (ISCC), Gammarth, Tunisia, 2023, pp. 950-955, doi: 10.1109/ISCC58397.2023.10218254

[24] Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.S.; Davis, A.; Dean, J.; Devin, M.; et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. 2015. Available online: tensorflow.org (accessed on 3 January 2024).